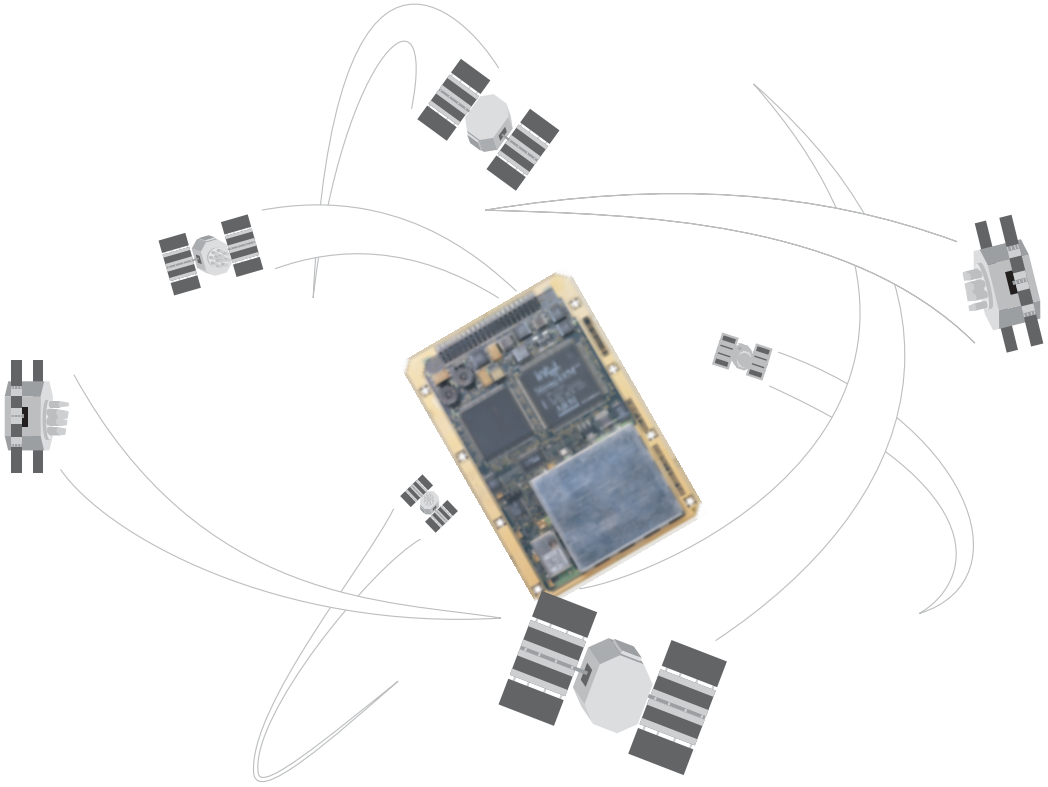
OEM4
Family of Receivers

USER MANUAL - VOLUME 2
Command and Log Reference



OEM4 Family of Receivers - Command and Log Reference Manual

Publication Number: OM-20000047
Revision Level: 8
Revision Date: 2002/05/02

This manual reflects Firmware Version 1.400

Proprietary Notice

Information in this document is subject to change without notice and does not represent a commitment on the part of NovAtel Inc. The software described in this document is furnished under a licence agreement or non-disclosure agreement. The software may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or non-disclosure agreement.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose without the express written permission of a duly authorized representative of NovAtel Inc.

The information contained within this manual is believed to be true and correct at the time of publication.

NovAtel, GPSolution, MiLLennium, ProPak, RT-20 and RT-2 are registered trademarks of NovAtel Inc.

PAC, GPSCard and GPSAntenna are trademarks of NovAtel Inc.

All other brand names are trademarks of their respective holders.

© Copyright 2002 NovAtel Inc. All rights reserved.

Unpublished rights reserved under International copyright laws.

Printed in Canada on recycled paper. Recyclable.

Table of Contents

Proprietary Notice 2

Foreword 10

Congratulations! 10
Scope 10
Prerequisites 10
Conventions 10
What's New in Firmware Version 1.400 Since Version 1.200 11

1 Messages 12

1.1 Message Types 12
1.1.1 ASCII 13
1.1.2 Abbreviated ASCII 15
1.1.3 Binary 15
1.2 Responses 18
1.3 GPS Time Status 19
1.4 Message Time Stamps 20
1.5 32-Bit CRC 21

2 Commands 22

2.1 Command Formats 22
2.2 Command Settings 22
2.3 Commands by Function 23
2.4 MiLLennium GPSCard Compatibility 33
2.5 Factory Defaults 34
2.6 Command Reference 35
2.6.1 ADJUST1PPS 35
2.6.2 ANTENNAPOWER 38
2.6.3 ASSIGN 39
2.6.4 ASSIGNALL 41
2.6.5 ASSIGNOMNI 43
2.6.6 AUTH 44
2.6.7 CLOCKADJUST 46
2.6.8 CLOCKCALIBRATE 47
2.6.9 COM 49
2.6.10 COMCONTROL 51
2.6.11 CSMOOTH 52
2.6.12 DATUM 53
2.6.13 DGPSEPHEMDELAY DGPS 56
2.6.14 DGPSRXID 57
2.6.15 DGPSTIMEOUT DGPS 58
2.6.16 DGPSTXID DGPS 59
2.6.17 DYNAMICS 60
2.6.18 ECUTOFF 61
2.6.19 EXTERNALCLOCK 62
2.6.20 FIX 65
2.6.21 FREQUENCYOUT 67

2.6.22 FRESET 68
2.6.23 INTERFACEMODE DGPS 69
2.6.24 LOCKOUT 70
2.6.25 LOG 71
2.6.26 MAGVAR 74
2.6.27 MODEL 76
2.6.28 NVMRESTORE 77
2.6.29 POSAVE 78
2.6.30 PSRDIFFSOURCE DGPS 79
2.6.31 RESET 81
2.6.32 RTKBASELINE RTK 82
2.6.33 RTKCOMMAND RTK 84
2.6.34 RTKDYNAMICS RTK 85
2.6.35 RTKELEVMASK RTK 86
2.6.36 RTKSOLUTION RTK 87
2.6.37 RTKSOURCE RTK 88
2.6.38 RTKSVENTRIES RTK 89
2.6.39 SAVECONFIG 90
2.6.40 SEND 91
2.6.41 SENDHEX 92
2.6.42 SETAPPROXPOS 93
2.6.43 SETAPPROXTIME 94
2.6.44 SETNAV 95
2.6.45 SETRTCM16 DGPS & RTK 96
2.6.46 STATUSCONFIG 97
2.6.47 UNASSIGN 98
2.6.48 UNASSIGNALL 98
2.6.49 UNDULATION 99
2.6.50 UNLOCKOUT 100
2.6.51 UNLOCKOUTALL 100
2.6.52 UNLOG 101
2.6.53 UNLOGALL 101
2.6.54 USERDATUM 102
2.6.55 WAASCORRECTION SBAS 103

3 Data Logs 104

3.1 Log Types 104
3.2 Logs By Function 105
3.3 MiLLennium GPSCard Compatibility 118
3.4 Log Reference 119
3.4.1 ALMANAC Decoded Almanac 119
3.4.2 AVEPOS Position Averaging 121
3.4.3 BESTPOS Best Position 122
3.4.4 BESTVEL Best Available Velocity Data 125
3.4.5 BESTXYZ Best Available Cartesian Position and Velocity 126
3.4.6 CLOCKMODEL Current Clock Model Status 128
3.4.7 CLOCKSTEERING Clock Steering Status 130
3.4.8 CMR Standard Logs RTK 132

3.4.9 CMRDATADESC Base Station Description RTK 133
3.4.10 CMRDATAOBS Base Station Satellite Observations RTK 135
3.4.11 CMRDATAREF Base Station Position RTK 137
3.4.12 COMCONFIG Current COM Port Configuration 139
3.4.13 GPALM Almanac Data NMEA 140
3.4.14 GPGGA GPS Fix Data and Undulation NMEA 141
3.4.15 GPGGARTK Global Position System Fix Data NMEA 142
3.4.16 GPGLL Geographic Position NMEA 143
3.4.17 GPGRS GPS Range Residuals for Each Satellite NMEA 144
3.4.18 GPGSA GPS DOP and Active Satellites NMEA 145
3.4.19 GPGST Pseudorange Measurement Noise Statistics NMEA 146
3.4.20 GPGSV GPS Satellites in View NMEA 147
3.4.21 GPRMB Navigation Information NMEA 148
3.4.22 GPRMC GPS Specific Information NMEA 149
3.4.23 GPSEPHEM Decoded GPS Ephemerides 150
3.4.24 GPVTG Track Made Good And Ground Speed NMEA 152
3.4.25 GPZDA UTC Time and Date NMEA 153
3.4.26 IONUTC Ionospheric and UTC Data 154
3.4.27 LOGLIST List of System Logs 155
3.4.28 MARKPOS Position at Time of MarkIn Event 157
3.4.29 MARKTIME Time of MarkIn Event 158
3.4.30 MATCHEDPOS Matched RTK Position RTK 159
3.4.31 MATCHEDXYZ Matched RTK Cartesian Position RTK 161
3.4.32 NAVIGATE User Navigation Data 162
3.4.33 NMEA Standard Logs 164
3.4.34 OMNIHPPOS OMNISTAR HP Position 165
3.4.35 OMNIINFO OmniSTAR Configuration Information 166
3.4.36 OMNISTAT OmniSTAR Status Information 168
3.4.37 PASSCOM1, PASSCOM2, PASSCOM3 Redirect Data 170
3.4.38 PORTSTATS COM Port Statistics 171
3.4.39 PSRDOP Pseudorange DOP 172
3.4.40 PSRPOS Pseudorange Position 173
3.4.41 PSRVEL Pseudorange Velocity 174
3.4.42 PSRXYZ Pseudorange Cartesian Position and Velocity 175
3.4.43 RANGE Satellite Range Information 177
3.4.44 RANGECMP Compressed Version of the RANGE Log 180
3.4.45 RAWALM Raw Almanac Data 182
3.4.46 RAWEPHEM Raw Ephemeris 183
3.4.47 RAWGPSSUBFRAME Raw Subframe Data 184
3.4.48 RAWGPSWORD Raw Navigation Word 185
3.4.49 RAWOMNIFRAME Raw OmniSTAR Frame Data 186
3.4.50 RAWOMNIPACKET Raw OmniSTAR Data Packet 187
3.4.51 RAWWAASFRAME Raw SBAS Frame Data SBAS 188
3.4.52 REFSTATION Base Station Position and Health RTK 189
3.4.53 RTCA Standard Logs DGPS 190
3.4.54 RTCADATA1 Differential GPS Corrections DGPS 191
3.4.55 RTCADATAEPHEM Ephemeris and Time Information DGPS & RTK192
3.4.56 RTCADATAOBS Base Station Observations RTK 193

3.4.57 RTCADATAREF Base Station Parameters RTK 195
3.4.58 RTCM Standard Logs DGPS 196
3.4.59 RTCMDATA1 Differential GPS Corrections DGPS 197
3.4.60 RTCMDATA3 Base Station Parameters RTK 198
3.4.61 RTCMDATA9 Partial Differential GPS Corrections DGPS 199
3.4.62 RTCMDATA15 Ionospheric Corrections DGPS 200
3.4.63 RTCMDATA16 Special Message DGPS & RTK 201
3.4.64 RTCMDATA1819 Raw Measurements RTK 202
3.4.65 RTCMDATA2021 Measurement Corrections RTK 206
3.4.66 RTCMDATA22 Extended Base Station RTK 209
3.4.67 RTCMDATA59 Type 59N-0 NovAtel RT20 Differential RTK 210
3.4.68 RTKDATA RTK Solution Parameters RTK 212
3.4.69 RTKPOS RTK Low Latency Position Data RTK 214
3.4.70 RTKVEL RTK Velocity RTK 215
3.4.71 RTKXYZ RTK Cartesian Position and Velocity RTK 216
3.4.72 RXCONFIG Receiver Configuration 218
3.4.73 RXHWLEVELS Receiver Hardware Levels 219
3.4.74 RXSTATUS Receiver Status 220
3.4.75 RXSTATUSEVENT Status Event Indicator 225
3.4.76 SATVIS Satellite Visibility 226
3.4.77 SATXYZ SV Position in ECEF Cartesian Coordinates 227
3.4.78 TIME Time Data 228
3.4.79 TIMESYNC Synchronize Time Between GPS Receivers 229
3.4.80 TRACKSTAT Tracking Status 230
3.4.81 VALIDMODELS Valid Model Information 232
3.4.82 VERSION Version Information 233
3.4.83 WAAS0 Remove PRN from Solution SBAS 235
3.4.84 WAAS1 PRN Mask Assignments SBAS 236
3.4.85 WAAS2 Fast Correction Slots 0-12 SBAS 237
3.4.86 WAAS3 Fast Corrections Slots 13-25 SBAS 239
3.4.87 WAAS4 Fast Correction Slots 26-38 SBAS 240
3.4.88 WAAS5 Fast Correction Slots 39-50 SBAS 241
3.4.89 WAAS6 Integrity Message SBAS 242
3.4.90 WAAS7 Fast Correction Degradation SBAS 244
3.4.91 WAAS9 GEO Navigation Message SBAS 246
3.4.92 WAAS10 Degradation Factor SBAS 247
3.4.93 WAAS12 SBAS Network Time and UTC SBAS 248
3.4.94 WAAS17 GEO Almanac Message SBAS 249
3.4.95 WAAS18 IGP Mask SBAS 250
3.4.96 WAAS24 Mixed Fast/Slow Corrections SBAS 251
3.4.97 WAAS25 Long-Term Slow Satellite Corrections SBAS 253
3.4.98 WAAS26 Ionospheric Delay Corrections SBAS 255
3.4.99 WAAS27 SBAS Service Message SBAS 256
3.4.100 WAASCORR SBAS Range Corrections Used SBAS 257

4 Responses 258

Tables

1 Field Types 12
2 Byte Arrangements 13
3 ASCII Message Header Structure 14
4 Binary Message Header Structure 16
5 Serial Port Identifiers 17
6 GPS Time Status 19
7 Commands By Function Table 23
8 OEM4 Family Commands in Alphabetical Order 27
9 OEM4 Family Commands in Order of their Message IDs 30
10 OEM4 Family Command Comparison 33
11 Channel State 39
12 Channel System 42
13 Serial Port Identifiers 50
14 Parity 50
15 Handshaking 50
16 Reference Ellipsoid Constants 54
17 Datum Transformation Parameters 54
18 User Dynamics 60
19 Clock Type 64
20 Default Values of Process Noise Elements 64
21 FIX Parameters 65
22 FIX Types 66
23 FRESET Target 68
24 Serial Port Interface Modes 69
25 LOG Command Binary Format 72
26 LOG Command ASCII Format 73
27 DGPS Type 80
28 Baseline Parameters 82
29 Baseline Type 83
30 Dynamics Mode 85
31 Type of Carrier Phase Ambiguity 87
32 Mask Types 97
33 Log Type Triggers 104
34 Logs By Function Table 105
35 OEM4 Family Logs in Alphabetical Order 110
36 OEM4 Family Logs in Order of their Message IDs 114
37 MiLLennium OEM3 Log Comparison 118
38 Position Averaging Status 121
39 Position or Velocity Type 123
40 Solution Status 124
41 Clock Model Status 128
42 Clock Source 131
43 Steering State 131
44 Position Accuracy 138
45 Position Precision of NMEA Logs 143
46 URA Variance 151

47 LOGLIST Binary Format 155
48 LOGLIST ASCII Format 156
49 Navigation Data Type 162
50 OmniSTAR Subscription Type 167
51 OmniSTAR Signal Tracking Status 169
52 VBS Status Word 169
53 Channel Tracking Status 177
54 Channel Tracking Example 178
55 Tracking State 179
56 Correlator Spacing 179
57 Range Record Format (RANGECMP only) 181
58 Base Station Status 189
59 Base Station Type 189
60 RTCM1819 Data Quality Indicator 204
61 RTCM1819 Smoothing Interval 205
62 RTCM1819 Multipath Indicator 205
63 RTCM2021 Data Quality Indicator 208
64 RTCM2021 Multipath Indicator 208
65 Searcher Type 213
66 Ambiguity Types 213
67 RTK Information 213
68 Receiver Hardware Parameters 219
69 Receiver Error 222
70 Receiver Status 223
71 Auxiliary 1 Status 224
72 Auxiliary 2 Status 224
73 Auxiliary 3 Status 224
74 Status Word 225
75 Event Type 225
76 Range Reject Codes 231
77 Component Types 234
78 Version Log: Field Formats 234
79 Evaluation of UDREI 238
80 Response Messages 258

Figures

1 1PPS Alignment 35
2 Illustration of Magnetic Variation & Correction 75
3 Using the SEND Command 91
4 Illustration of SETNAV Parameters 95
5 Illustration of Undulation 99
6 The WGS84 ECEF Coordinate System 127
7 Navigation Parameters 162
8 1PPS Alignment-TIMESYNC Log 229

Foreword

Congratulations!

Congratulations on purchasing a NovAtel product.

Whether you have bought a stand alone GPSCard or a packaged receiver you will have also received companion documents to this manual. They will help you get the hardware operational. Afterwards, this text will be your primary OEM4 family command and logging reference.

Scope

This manual describes each command and log that the OEM4 family of receivers are capable of accepting or generating. Sufficient detail is provided so that you should understand the purpose, syntax, and structure of each command or log and be able to effectively communicate with the card, thus enabling you to effectively use and write custom interfacing software for specific needs and applications. The manual is organized into chapters which allow easy access to appropriate information about the receiver.

There is also Satellite Based Augmentation System (SBAS) signal functionality. Please see the *SBAS Overview* appendix in *Volume 1* of this manual and the *Conventions* section below for more information.

This manual does not address any of the receiver hardware attributes or installation information. Please consult *Volume 1* of this manual for technical information on these topics. Furthermore, should you encounter any functional, operational, or interfacing difficulties with the receiver, consult *Volume 1* of this manual for NovAtel warranty and customer support information.

Prerequisites

As this reference manual is focused on the OEM4 family commands and logging protocol, it is necessary to ensure that the card has been properly installed and powered up according to the instructions outlined in the companion OEM4 User Manual Volume 1 before proceeding.

Conventions

This manual covers the full performance capabilities of all OEM4 family of receivers. Feature-tagging symbols have been created to help clarify which commands and logs are only available with the RT-2 or RT-20 option or if there is only partial implementation of this feature:

RTK	Features available only with GPSCards equipped with the RT-20 or RT-2 option
DGPS	Feature used when operating in differential mode
NMEA	National Marine Electronics Association format, see *Message Formats* in *Volume 1* of this manual
SBAS	SBAS messages can be generated if you have an SBAS capable receiver model and are tracking an SBAS satellite. For more information see the *SBAS Overview* appendix in *Volume 1* of this manual.

Other simple conventions are:

H	The letter H in the Binary Byte or Binary Offset columns of the commands and logs tables represents the header length for that command or log, see *Binary on Page 15*
0x	The number following 0x is a hexadecimal number

When default values are shown in command tables, they indicate the assumed values when optional parameters have been omitted. Default values do not imply the factory default settings, see *Chapter 2, Page 34* for a list of factory default settings.

Command descriptions use the bracket symbols, [ ], to represent the optionality of parameters.

In tables where values are missing they should be assumed to be reserved for future use.

Status words are output as hexadecimal numbers and must be converted to binary format (and in some cases then also to decimal). For an example of this type of conversion, please see the RANGE log, *Table 54, Channel Tracking Example, on Page 178*. Conversions and their binary or decimal results are always read from right to left. For a complete list of hexadecimal, binary and decimal equivalents, please see the *Unit Conversion* appendix in *Volume 1* of this manual.

ASCII log examples may be split over several lines for readability. In reality only a single [CR][LF] pair is transmitted at the end of an ASCII log.

The terms Euro4 and OEM4 will not be used in this manual unless a specific detail refers to it alone. The term receiver will infer that the text is applicable to an OEM4 or EURO4, either stand-alone or in an enclosure, unless otherwise stated.

All of the relevant SBAS commands and logs start with WAAS except for RAWWAASFRAME.

Generally, the PRN field of the WAASx logs is common, and indicates the SBAS satellite that the message originated from.

Please refer to the RTCA document *RTCA D0-229B, Appendix A Wide Area Augmentation System Signal Specification* for detail on the SBAS logs.

What's New in Firmware Version 1.400 Since Version 1.200

1. Commands and logs have been added for use with an OmniSTAR-compatible OEM4 family receiver. See ASSIGNOMNI *on Page 43*, OMNIHPPOS *on Page 165*, OMNIINFO *on Page 166*, OMNISTAT *on Page 168*, RAWOMNIFRAME *on Page 186* and RAWOMNIPACKET *on Page 187*.
2. The DGPSRXID command has been replaced by the PSRDIFFSOURCE command *on Page 79* and the RTKSOURCE command *on Page 88*.
3. Additional receiver clock and time synchronization commands and logs have been added. See ADJUST1PPS *on Page 35*, CLOCKCALIBRATE *on Page 47*, CLOCKSTEERING *on Page 130* and TIMESYNC *on Page 229*.
4. A new command has been added, see COMCONTROL *on Page 51*, to manage the hardware control lines of RS232 ports.
5. A new field for the issue of ionospheric data (iodi) has been added to the WAAS26 log, see *Page 255*. Also, as a correction in this manual, additional slot number fields have been added in the WAAS2-WAAS 7 and WAAS24 logs, see *Pages 237-244* and *Page 251* respectively.

Chapter 1 Messages

1.1 Message Types

The receiver handles all incoming and outgoing NovAtel data in three different message formats: Abbreviated ASCII, ASCII, and Binary. This allows for a great deal of versatility in the way the OEM4 can be used. All NovAtel commands and logs can be entered, transmitted, output or received in any of the three formats. The receiver also supports RTCA, RTCM, CMR and NMEA format messaging, see the chapter on *Message Formats* in *Volume 1* of this manual.

When entering an ASCII or abbreviated ASCII command in order to request an output log, the message type is indicated by the character appended to the end of the message name. 'A' indicates that the message is ASCII and 'B' indicates that it is binary. No character means that the message is Abbreviated ASCII. When issuing binary commands the output message type is dependant on the bit format in the message's binary header, see *Binary on Page 15*.

The following table describes the field types used in the description of messages.

Table 1: Field Types

Type	Binary Size (bytes)	Description
Char	1	The **char** type is an 8-bit integer in the range -128 to +127. This integer value may be the ASCII code corresponding to the specified character. In ASCII or Abbreviated ASCII this comes out as an actual character.
UChar	1	The **uchar** type is an 8-bit unsigned integer. Values are in the range from +0 to +255. In ASCII or Abbreviated ASCII this comes out as a number.
Short	2	The short type is 16-bit integer in the range -32768 to +32767.
UShort	2	The same as Short except that it is not signed. Values are in the range from +0 to +65535.
Long	4	The **long** type is 32-bit integer in the range -2147483648 to +2147483647.
ULong	4	The same as Long except that it is not signed. Values are in the range from +0 to +4294967295.
Double	8	The **double** type contains 64 bits: 1 for sign, 11 for the exponent, and 52 for the mantissa. Its range is ±1.7E308 with at least 15 digits of precision. This is IEEE 754.
Float	4	The **float** type contains 32 bits: 1 for the sign, 8 for the exponent, and 23 for the mantissa. Its range is ±3.4E38 with at least 7 digits of precision. This is IEEE 754.
Enum	4	A 4-byte enumerated type beginning at zero (an unsigned long). In binary, the enumerated value is output. In ASCII or Abbreviated ASCII, the enumeration label is spelled out.
GPSec	4	This type has two separate formats that depend on whether you have requested a binary or an ASCII format output. For binary the output is in milliseconds and is a **long** type. For ASCII the output is in seconds and is a **float** type.
Hex	n	Hex is a packed, fixed length (n) array of bytes in binary but in ASCII or Abbreviated ASCII is converted into 2 character hexadecimal pairs.
String	n	String is a variable length array of bytes that is null-terminated in the binary case. The maximum byte length for each String field is shown in their row in the log or command tables.

Table 2: Byte Arrangements

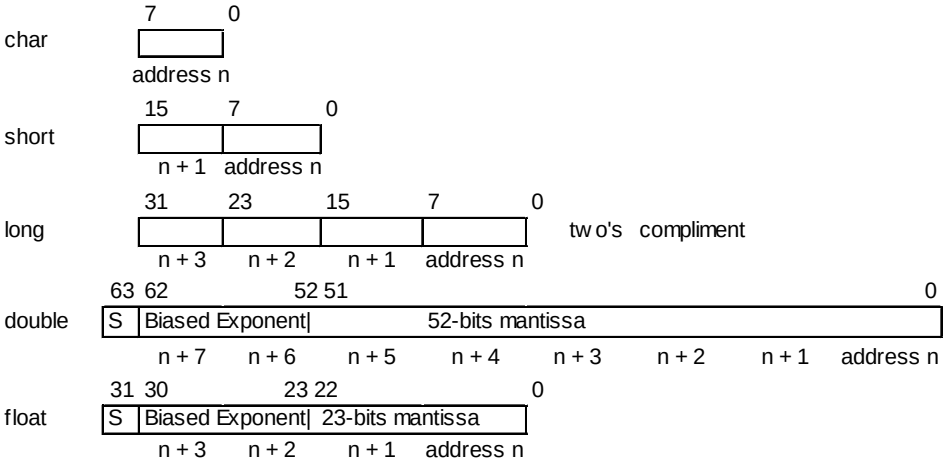

char
7 0
address n
short
15 7 0
n + 1 address n
long
31 23 15 7 0
two's compliment
n + 3 n + 2 n + 1 address n
double
63 62 52 51 0
S Biased Exponent| 52-bits mantissa
n + 7 n + 6 n + 5 n + 4 n + 3 n + 2 n + 1 address n
float
31 30 23 22 0
S Biased Exponent| 23-bits mantissa
n + 3 n + 2 n + 1 address n

Table 2 shows the arrangement of bytes within each field type when used by IBM PC computers. All data sent to or from the OEM4 card, however, is read least significant bit (LSB) first, opposite to what is shown in *Table 2*. Data is then stored in the OEM4 card LSB first. For example, in char type data, the LSB is bit 0 and the most significant bit (MSB) is bit 7. See *Table 54, Channel Tracking Example on Page 178* for a more detailed example.

1.1.1 ASCII

ASCII messages are readable by both the user and a computer. The structures of all ASCII messages follow the general conventions as noted here:

1. The lead code identifier for each record is '#'.
2. Each log or command is of variable length depending on amount of data and formats.
3. All data fields are delimited by a comma ',' with two exceptions. The first exception is the last header field which is followed by a ; to denote the start of the data message. The other exception is the last data field, which is followed by a * to indicate end of message data.
4. Each log ends with a hexadecimal number preceded by an asterisk and followed by a line termination using the carriage return and line feed characters, e.g., *1234ABCD[CR][LF]. This value is a 32-bit CRC of all bytes in the log, excluding the '#' identifier and the asterisk preceding the four checksum digits. See *32-Bit CRC on Page 21* for the algorithm used to generate the CRC.
5. An ASCII string is one field and is surrounded by double quotation marks, e.g. "ASCII string". If separators are surrounded by quotation marks then the string is still one field and the separator will be ignored, e.g. "xxx,xxx" is one field. Double quotation marks within a string are not allowed.
6. If the card detects an error parsing an input message, it will return an error response message. Please see *Chapter 4, Responses on Page 258* for a list of response messages from the receiver.

Message Structure:

header;	data field...,	data field...,	data field...	*xxxxxxxx	[CR][LF]

The ASCII message header is formatted as follows:

Table 3: ASCII Message Header Structure

Field #	Field Name	Field Type	Description	Ignored on Input
1	Sync	Char	Sync character. The ASCII message is always preceded by a single '#' symbol.	N
2	Message	Char	This is the ASCII name of the log (see a list of all the logs in *Table 34, Logs By Function Table on Page 105*).	N
3	Port	Char	This is the name of the port from which the log was generated. The string is made up of the port name followed by an _x where x is a number from 1 to 31 denoting the virtual address of the port. If no virtual address is indicated, it is assumed to be address 0.	Y
4	Sequence #	Long	This is used for multiple related logs. It is a number that counts down from N-1 to 0 where 0 means it is the last one of the set. Most logs only come out one at a time in which case this number is 0.	N
5	Idle Time	Float	The minimum percentage of time that the processor is idle.	Y
6	GPS Time Status	Enum	This value indicates the quality of the GPS time (see *Table 6, GPS Time Status on Page 19*)	Y
7	Week	Ulong	GPS week number.	Y
8	Seconds	GPSec	Seconds from the beginning of the GPS week accurate to the millisecond level.	Y
9	Receiver Status	Ulong	This is an eight digit hexadecimal number representing the status of various hardware and software components of the receiver (see *Table 70, Receiver Status on Page 223*)	Y
10	Reserved	Ulong	Reserved for internal use.	Y
11	Receiver s/w Version	Ulong	This is a value (0 - 65535) that represents the receiver software build number.	Y
12	;	Char	This character indicates the end of the header.	N

Example Log:

```
#RAWEPHEMA,COM1,0,70.0,SATTIME,1023,512190.000,00000000,2e9f,0;
7,1023,518400,1023,512190.083,8b0ffca6bb24ffd70173931172f3b03c9c99c678fc
267e9000ffe14b88a0,8b0ffca6b92b26087a3187cbf13a9e079605828cd00ee8a10d735
27e907e,8b0ffca6b9ad0088f0526ba5005a26e038131e06a8925200ffa76826fcd8*2dd
cf8b5[CR][LF]
```

1.1.2 Abbreviated ASCII

This message format is designed to make the entering and viewing of commands and logs by the user as simple as possible. The data is represented as simple ASCII characters separated by spaces or commas and arranged in an easy to understand fashion. There is also no 32-bit CRC for error detection because it is meant for viewing by the user.

Example Command:

```
log com1 loglist
```

Resultant Log:

```
<LOGLIST COM1 0 69.0 FINE 0 0.000 00240000 206d 0
<     4
<          COM1 RXSTATUSEVENTA ONNEW 0.000000 0.000000 NOHOLD
<          COM2 RXSTATUSEVENTA ONNEW 0.000000 0.000000 NOHOLD
<          COM3 RXSTATUSEVENTA ONNEW 0.000000 0.000000 NOHOLD
<          COM1 LOGLIST ONCE 0.000000 0.000000 NOHOLD
```

As you can see the array of 4 logs are offset from the left hand side and start with '<'.

1.1.3 Binary

Binary messages are meant strictly as a machine readable format. They are also ideal for applications where the amount of data being transmitted is fairly high. Because of the inherent compactness of binary as opposed to ASCII data, the messages are much smaller. This allows a larger amount of data to be transmitted and received by the receiver's communication ports. The structure of all Binary messages follows the general conventions as noted here:

1. Basic format of:

Header	3 Sync bytes plus 25 bytes of header information. The header length is variable as fields may be appended in the future. Always check the header length.
Data	variable
CRC	4 bytes

2. The 3 Sync bytes will always be:

Byte	Hex	Decimal
First	AA	170
Second	44	68
Third	12	18

3. The CRC is a 32-bit CRC (see *32-Bit CRC on Page 21* for the CRC algorithm) performed on all data including the header.

4. The header is in the format shown in *Table 4, Binary Message Header Structure on Page 16*.

Table 4: Binary Message Header Structure

Field #	Field Name	Field Type	Description	Binary Bytes	Binary Offset	Ignored on Input
1	Sync	Char	Hexadecimal 0xAA.	1	0	N
2	Sync	Char	Hexadecimal 0x44.	1	1	N
3	Sync	Char	Hexadecimal 0x12.	1	2	N
4	Header Length	Uchar	Length of the header.	1	3	N
5	Message ID	Ushort	This is the Message ID number of the log (see the log descriptions in *Table 36, OEM4 Family Logs in Order of their Message IDs on Page 114* for the Message ID values of individual logs).	2	4	N
6	Message Type	Char	Bits 0-4 = Reserved Bits 5-6 = Format 00 = Binary 01 = ASCII 10 = Abbreviated ASCII, NMEA 11 = Reserved Bit 7 = Response Bit (see *Section 1.2, Responses on Page 18*) 0 = Original Message 1 = Response Message	1	6	N
7	Port Address	Char	See *Table 5, Serial Port Identifiers on Page 17* (lower 8 bits only)	1	7	N [a]
8	Message Length	Ushort	The length in bytes of the body of the message. This does not include the header nor the CRC.	2	8	N
9	Sequence	Ushort	This is used for multiple related logs. It is a number that counts down from N-1 to 0 where N is the number of related logs and 0 means it is the last one of the set. Most logs only come out one at a time in which case this number is 0.	2	10	N
10	Idle Time	Char	The percentage of time that the processor is idle in the last second. Take the time (0 - 200) and divide by two to give the percentage of time (0 - 100%).	1	12	Y
11	Time Status	Enum	Indicates the quality of the GPS time (see *Table 6, GPS Time Status on Page 19*).	1 [b]	13	N [c]
12	Week	Ushort	GPS week number.	2	14	N [c]
13	Milliseconds	GPSec	Milliseconds from the beginning of the GPS week.	4	16	N [c]
14	Receiver Status	Ulong	32 bits representing the status of various hardware and software components of the receiver (see *Table 70, Receiver Status on Page 223*)	4	20	Y
15	Reserved	Ushort	Reserved for internal use.	2	24	Y
16	Receiver S/W Version	Ushort	This is a value (0 - 65535) that represents the receiver software build number.	2	26	Y

a. Recommended value is THISPORT (binary 192)

b. This ENUM is not 4 bytes long but, as indicated in the table, is only 1 byte.

c. These time fields are ignored if Field #11, Time Status, is invalid. In this case the current receiver time is used. The recommended values for the three time fields are 0, 0, 0.

Table 5: Serial Port Identifiers

ASCII Port Name	Port Value	Description
NO_PORTS	0	No ports specified
COM1_ALL	1	All virtual ports for COM port 1
COM2_ALL	2	All virtual ports for COM port 2
COM3_ALL	3	All virtual ports for COM port 3
THISPORT_ALL	6	All virtual ports for the current COM port
ALL_PORTS	8	All virtual ports for all COM ports
COM1	32	COM port 1, virtual port 0
COM1_1	33	COM port 1, virtual port 1
. . .		
COM1_31	63	COM port 1, virtual port 31
COM2	64	COM port 2, virtual port 0
. . .		
COM2_31	95	COM port 2, virtual port 31
COM3	96	COM port 3, virtual port 0
. . .		
COM3_31	127	COM port 3, virtual port 31
THISPORT	192	Current COM port, virtual port 0
. . .		
THISPORT_31	223	Current COM port, virtual port 31

✉ COM1_ALL, COM2_ALL, COM3_ALL, THISPORT_ALL and ALL_PORTS are only valid for the UNLOGALL command, see *Page 101*.

1.2 Responses

By default, if you input a message you will get back a response. If desired, the INTERFACEMODE command can be used to disable response messages (see *Page 69*). The response will be in the exact format that you entered the message (i.e. binary input = binary response).

Abbreviated Response

Just the leading '<' followed by the response string:

e.g. <OK

ASCII Response

Full header with the message name being identical except ending in an 'R' (for response). The body of the message consists of a 40 character string for the response string.

e.g. #BESTPOSR,COM1,0,67.0,FINE,1028,422060.400,00000000,a31b,0;"OK" *b867caad

Binary Response

Similar to an ASCII response except that it follows the binary protocols:

- Binary header with message type set to response value (0x80), see *Field 6 in Table 4, Binary Message Header Structure on Page 16.*
- ENUM response ID, see *Table 80, Response Messages on Page 258.*
- Binary response string to match the ENUM response ID above.

1.3 GPS Time Status

All reported receiver times are subject to a qualifying time status. This status gives you an indication of how well a time is known, see *Table 6*:

Table 6: GPS Time Status[a]

GPS Time Status (Decimal)	GPS Time Status (ASCII)	Description
20	UNKNOWN	Time validity is unknown.
60	APPROXIMATE	Time is set approximately.
80	COARSEADJUSTING	Time is approaching coarse precision.
100	COARSE	This time is valid to coarse precision.
120	COARSESTEERING	Time is coarse set, and is being steered.
130	FREEWHEELING	Position is lost, and the range bias cannot be calculated.
140	FINEADJUSTING	Time is adjusting to fine precision.
160	FINE	Time has fine precision.
180	FINESTEERING	Time is fine, set and is being steered.
200	SATTIME	Time from satellite. This is only used in logs containing satellite data such as ephemeris and almanac.

a. See also *Section 1.4, Message Time Stamps on Page 20.*

There are several distinct states that the card will go through:

- UNKNOWN
- COARSE
- FREEWHEELING
- FINE
- FINESTEERING

On start up, and before any satellites are being tracked, the receiver can not possibly know the current time. As such, the receiver time starts counting at GPS week 0 and second 0.0. The time status flag is set to UNKNOWN.

If time is input to the receiver using the SETAPPROXTIME command, see *Page 94*, or on receipt of an RTCAEPHEM message, see *Page 150*, the time status will be APPROXIMATE.

After the first ephemeris is decoded, the receiver time is set to a resolution of ±10 milliseconds. This state is qualified by the COARSE or COARSESTEERING time status flag depending on the state of the CLOCKADJUST switch.

Once a position is known and range biases are being calculated, the internal clock model will begin modelling the position range biases and the receiver clock offset.

Modelling will continue until the model is a good estimation of the actual receiver clock behavior. At this time, the receiver time will again be adjusted, this time to an accuracy of ±1 microsecond. This state is qualified by the FINE time status flag.

The final logical time status flag depends on whether CLOCKADJUST is enabled or not, refer to the *CLOCKADJUST* command in Volume 2 of this manual. If CLOCKADJUST is disabled, the time status flag will never improve on FINE. The time will only be adjusted again to within ±1 microsecond if the range bias gets larger than ±250 milliseconds. If ClockAdjust is enabled, the time status flag will be set to FINE_STEERING and the receiver time will be continuously updated (steered) to minimize the receiver range bias.

If for some reason position is lost and the range bias cannot be calculated, the time status will be degraded to FREEWHEELING.

1.4 Message Time Stamps

All NovAtel format messages generated by the OEM4 have a GPS time stamp in their header. GPS time is referenced to UTC with zero point defined as midnight on the night of January 5 1980. The time stamp consists of the number of weeks since that zero point and the number of seconds since the last week number change (0 to 603,799). GPS time differs from UTC time since leap seconds are occasionally inserted into UTC but GPS time is continuous. In addition a small error (less than 1 microsecond) can exist in synchronization between UTC and GPS time. The OEM4 TIME log reports both GPS and UTC time and the offset between the two.

The data in synchronous logs (e.g. RANGE, BESTPOS, TIME) are based on a periodic measurement of satellite pseudoranges. The time stamp on these logs is the receiver estimate of GPS time at the time of the measurement. When setting time in external equipment, a small synchronous log with a high baud rate will be accurate to a fraction of a second. A synchronous log with trigger ONTIME 1 can be used in conjunction with the 1PPS signal to provide relative accuracy better than 250 ns.

Other log types (asynchronous and polled) are triggered by an external event and the time in the header may not be synchronized to the current GPS time. Logs that contain satellite broadcast data (e.g. ALMANAC, GPSEPHEM) have the transmit time of their last subframe in the header. In the header of differential time matched logs (e.g. MATCHEDPOS) is the time of the matched reference and local observation that they are based on. Logs triggered by a mark event (e.g. MARKEDPOS, MARKTIME) have the estimated GPS time of the mark event in their header. In the header of polled logs (e.g. LOGLIST, PORTSTATS, VERSION) is the approximate GPS time when their data was generated.

1.5 32-Bit CRC

The ASCII and Binary OEM4 message formats all contain a 32-bit CRC for data verification. This allows the user to ensure that the data received (or transmitted) is valid with a high level of certainty. This CRC can be generated using the following C algorithm:

```
#define CRC32_POLYNOMIAL   0xEDB88320L

/* --------------------------------------------------------------------------
Calculate a CRC value to be used by CRC calculation functions.
-------------------------------------------------------------------------- */
unsigned long CRC32Value(int i)
{
  int j;
  unsigned long ulCRC;

  ulCRC = i;
  for ( j = 8 ; j > 0; j-- )
  {
    if ( ulCRC & 1 )
      ulCRC = ( ulCRC >> 1 ) ^ CRC32_POLYNOMIAL;
    else
      ulCRC >>= 1;
  }
  return ulCRC;
}

/* --------------------------------------------------------------------------
Calculates the CRC-32 of a block of data all at once
-------------------------------------------------------------------------- */
unsigned long CalculateBlockCRC32(
  unsigned long ulCount,     /* Number of bytes in the data block */
  unsigned char *ucBuffer ) /* Data block */
{
  unsigned long ulTemp1;
  unsigned long ulTemp2;
  unsigned long ulCRC = 0;

  while ( ulCount-- != 0 )
  {
    ulTemp1 = ( ulCRC >> 8 ) & 0x00FFFFFFL;
    ulTemp2 = CRC32Value( ((int) ulCRC ^ *ucBuffer++ ) & 0xff );
    ulCRC = ulTemp1 ^ ulTemp2;
  }
  return( ulCRC );
}
```

Chapter 2 Commands

2.1 Command Formats

The receiver will accept commands in 3 formats as described in *Chapter 1*:

Abbreviated ASCII

ASCII

Binary

Abbreviated ASCII is the easiest to use for your input. The other two formats include a CRC for error checking and are intended for use when interfacing with other electronic equipment.

Here are examples of the same command in each format:

Abbreviated ASCII Example:

LOG COM2 BESTPOSB ONTIME 1[CR]

ASCII Example:

LOGA,COM2,0,66.0,UNKNOWN,0,15.917,004c0000,5255,32858;COM1,BESTPOSB, ONTIME,1.000000,0.000000,NOHOLD*F95592DD[CR]

Binary Example:

AA44121C 01000240 20000000 1D1D0000 29160000 00004C00 55525A80 20000000 2A000000 02000000 00000000 0000F03F 00000000 00000000 00000000 EC58E065

2.2 Command Settings

Their are several ways to determine the current command settings of the receiver:

1. Request an RXCONFIG log, see *Page 218*. This will provide a listing of all commands and their parameter settings. This log provides the most complete information, but the size and format do not make it easy to read.
2. For some specific commands, logs are available to indicate all their parameter settings. The LOGLIST log, see *Page 155*, will show all active logs in the receiver beginning with the LOG command. The COMCONFIG log, see *Page 135*, will show both the COM and INTERFACEMODE commands parameter settings for all serial ports.
3. Request a log of the specific command of interest. This will show the parameters last entered for that command. The format of the log produced is exactly the same as the format of the specific command with updated header information.

   This is very useful for most commands, but for commands that are repeated with different parameters (for example, COM, LOG, and INTERFACEMODE), this will only show the most recent set of parameters used. To see all sets of parameters try method 1 or 2 above.

Abbreviated ASCII Example:

```
log fix
<FIX COM1 0 45.0 FINE 1114 151898.288 00200000 dbfd 33123
<     NONE -10000.00000000000 -10000.00000000000 -10000.0000
```

2.3 Commands by Function

Table 7 lists the commands by function while *Table 8 on Page 27* is an alphabetical listing of commands (repeated in *Table 9 on Page 30* with the commands in the order of their message IDs). Please see *2.6, Command Reference on Page 35* for a more detailed description of individual commands which are listed alphabetically.

Table 7: Commands By Function Table

COMMUNICATIONS, CONTROL AND STATUS	
Commands	**Descriptions**
ANTENNAPOWER	Control power to low-noise amplifier (LNA) of an active antenna
COM	Set COM port configuration
COMCONTROL	Control the hardware control lines of the RS232 ports
FREQUENCYOUT	Set the output pulse train available on VARF
INTERFACEMODE	Set interface type, Receive (Rx)/Transmit (Tx), for a COM port
LOG	Request a log
SEND	Send ASCII message to COM port
SENDHEX	Send non-printable characters to a COM port
SETRTCM16	Enter ASCII message to be sent in RTCM data stream
UNLOG, UNLOGALL	Remove one or all logs from logging control

GENERAL RECEIVER CONTROL	
Commands	**Descriptions**
AUTH	Add authorization code for new model
DYNAMICS	Tune receiver parameters
RESET	Perform a hardware reset
FRESET	Reset receiver to factory default
MODEL	Switch receiver to a previously AUTHed model
NVMRESTORE	Restore NVM data after a failure in NVM
SAVECONFIG	Save current configuration

POSITION, PARAMETERS, AND SOLUTION FILTERING CONTROL	
Commands	**Descriptions**
CSMOOTH	Set amount of carrier smoothing
DATUM	Choose a DATUM name type
ECUTOFF	Set satellite elevation cut-off for solutions
FIX	Constrain receiver height, position or velocity
RTKBASELINE	Initialize RTK with a static baseline
RTKCOMMAND	Reset the RTK filter or set the filter to default settings
RTKDYNAMICS	Setup the RTK dynamics mode
RTKELEVMASK	Set the minimum elevation mask angle for satellites to include in RTK corrections
RTKSOLUTION	Set RTK carrier phase ambiguity type (Float or Fixed)
UNDULATION	Set ellipsoid-geoid separation
USERDATUM	Set user-customized datum
WAASCORRECTION	Set SBAS test mode and PRN

SATELLITE TRACKING AND CHANNEL CONTROL	
Commands	**Descriptions**
ASSIGN	Assign individual satellite channel
ASSIGNALL	Assign all satellite channels
DYNAMICS	Tune receiver parameters
ECUTOFF	Set satellite tracking elevation cut-off
SETAPPROXPOS	Set an approximate position
SETAPPROXTIME	Set an approximate GPS time
UNASSIGN	Unassign a previously ASSIGNed channel
UNASSIGNALL	Unassign all previously ASSIGNed channels

WAYPOINT NAVIGATION	
Commands	**Descriptions**
MAGVAR	Set magnetic variation correction
SETNAV	Set waypoints

DIFFERENTIAL BASE STATION	
Commands	**Descriptions**
DGPSEPHEMDELAY	DGPS ephemeris delay
DGPSTXID	DGPS transmit ID
FIX	Constrain receiver height, position or velocity
INTERFACEMODE	Set interface type Transmit (Tx), for a COMn port
LOG	Select required differential-output log
POSAVE	Set up position averaging
RTKELEVMASK	Set the minimum elevation mask angle for satellites to include in RTK corrections
RTKSVENTRIES	Set the number of satellites to include in RTK corrections

DIFFERENTIAL ROVER STATION	
Commands	**Descriptions**
ASSIGNOMNI	Set OmniSTAR satellite communication parameters
DGPSTIMEOUT	Set maximum age of differential data accepted
INTERFACEMODE	Set interface type, Receive (Rx), for a COMn port
PSRDIFFSOURCE	Set the pseudorange correction source
RTKDYNAMICS	Set the RTK dynamics mode
RTKBASELINE	Initialize RTK with a static baseline
RTKCOMMAND	Issue RTK specific commands
RTKELEVMASK	Set elevation mask to use for RTK positioning
RTKSOLUTION	Set RTK carrier phase ambiguity type (Float or Fixed) or disable
RTKSOURCE	Set the RTK correction source
SETAPPROXPOS	Set an approximate position
SETAPPROXTIME	Set an approximate GPS time

CLOCK INFORMATION, STATUS, AND TIME	
Commands	**Descriptions**
ADJUST1PPS	Adjust the receiver clock
CLOCKADJUST	Enable or disable adjustments to the internal clock and 1PPS output
CLOCKCALIBRATE	Adjust the control parameters of the clock steering loop
EXTERNALCLOCK	Set the parameters for an external clock (EURO4 only)
SETAPPROXTIME	Set an approximate time

Table 8: OEM4 Family Commands in Alphabetical Order

Command	Message ID	Description	Syntax
ADJUST1PPS	429	Adjust the receiver clock	adjust1pps mode [period] [offset]
ANTENNAPOWER	98	Control power to low-noise amplifier of an active antenna	antennapower flag
ASSIGN	27	Assign individual satellite channel to a PRN	assign channel [state] prn [Doppler [window]]
ASSIGNALL	28	Assign all satellite channels to a PRN	assignall [system] [state] prn [Doppler [window]]
ASSIGNOMNI	467	Set OmniSTAR satellite communication parameters	assignomni mode freq baud
AUTH	49	Add authorization code for new model	auth [state] part1 part2 part3 part4 part5 model [date]
CLOCKADJUST	15	Enable clock adjustments	clockadjust switch
CLOCKCALIBRATE	430	Adjust the control parameters of the clock steering loop	clockcalibrate mode [period] [width] [slope] [bandwidth]
COMCONTROL	431	Control the hardware control lines of the RS232 ports	comcontrol port signal control
COM	4	COM port configuration control	com [port] bps [parity [databits [stopbits [handshake [echo[break]]]]]]
CSMOOTH	269	Set carrier smoothing	csmooth L1time [L2time]
DATUM	160	Choose a DATUM name type	datum datum
DGPSEPHEMDELAY	142	DGPS ephemeris delay	dgpsephemdelay delay
DGPSTIMEOUT	127	Set maximum age of differential data accepted	dgpstimeout delay
DGPSTXID	144	DGPS transmit ID	dgpstxid type ID
DYNAMICS	258	Tune receiver parameters	dynamics dynamics
ECUTOFF	50	Set satellite elevation cut-off	ecutoff angle
EXTERNALCLOCK	230	Set external clock parameters	externalclock clocktype [freq] [h0 [h1 [h2]]]
FIX	44	Constrain to fixed height, position or velocity	fix type [param1 [param2 [param3]]]
FREQUENCYOUT	232	Sets the output pulse train available on VARF.	frequencyout [switch] [pulsewidth] [period]

Command	Message ID	Description	Syntax
FRESET	20	Clear almanac model, or user configuration data, which is stored in NVM and followed by a card reset.	freset [target]
INTERFACEMODE	3	Set interface type, Receive (Rx)/Transmit (Tx), for COM ports	interfacemode [port] rxtype txtype [responses]
LOCKOUT	137	Prevent the GPSCard from using a satellite by specifying its PRN	lockout prn
LOG	1	Request logs from receiver	log [port] message [trigger [period[offset[hold]]]]
MAGVAR	180	Set magnetic variation correction	magvar type [correction [stddev]]
MODEL	22	Switch to a previously AUTHed model	model model
NVMRESTORE	197	Restore NVM data after a failure in NVM	nvmrestore
POSAVE	173	Implement position averaging for base station	posave[state] maxtime[maxhstd[maxvstd]]
PSRDIFFSOURCE	493	Set the pseudorange correction source	psrdiffsource type ID
RESET	18	Perform a hardware reset	reset [delay]
RTKBASELINE	182	Initialize RTK with a static baseline	rtkbaseline type [par1 par2 par3 [2sigma]]
RTKCOMMAND	97	Reset the RTK filter or set the filter to default settings	rtkcommand action
RTKDYNAMICS	183	Set the RTK dynamics mode	rtkdynamics mode
RTKELEVMASK	91	Set the RTK mask angle	rtkelevmask type [angle]
RTKSOLUTION	184	Set RTK carrier phase ambiguity type (Float or Fixed) or disable	rtksolution type
RTKSOURCE	493	Set the RTK correction source	rtksource type ID
RTKSVENTRIES	92	Set the number of satellites to use in corrections	rtksventries number
SAVECONFIG	19	Save current configuration in non-volatile memory	saveconfig

Command	Message ID	Description	Syntax
SEND	177	Send an ASCII message to any of the communications ports	send port data
SENDHEX	178	Send non-printable characters in hexadecimal pairs	sendhex port length data
SETAPPROXPOS	377	Set an approximate position	setapproxpos lat lon height
SETAPPROXTIME	102	Set an approximate GPS time	setapproxtime week sec
SETNAV	162	Set start and destination waypoints	setnav fromlat fromlon tolat tolon track offset from-point to-point
SETRTCM16	131	Enter an ASCII text message to be sent out in the RTCM data stream	SETRTCM16 text
STATUSCONFIG	95	Configure various status mask fields in RXSTATUSEVENT log	statusconfig type word mask
UNASSIGN	29	Unassign a previously ASSIGNed channel	unassign channel
UNASSIGNALL	30	Unassign all previously ASSIGNed channels	unassignall [system]
UNDULATION	214	Choose undulation	undulation option [separation]
UNLOCKOUT	138	Reinstate a satellite in the solution computation	unlockout prn
UNLOCKOUTALL	139	Reinstate all previously locked out satellites	unlockoutall
UNLOG	36	Remove log from logging control	unlog [port] datatype
UNLOGALL	38	Remove all logs from logging control	unlogall [port]
USERDATUM	78	Set user-customized datum	userdatum semimajor flattening dx dy dz rx ry rz scale
WAASCORRECTION	312	Set SBAS test mode and PRN	waascorrection keyword [prn] [mode]

Table 9: OEM4 Family Commands in Order of their Message IDs

Message ID	Command	Description	Syntax
1	LOG	Request logs from receiver	log [port] message [trigger [period[offset[hold]]]]
3	INTERFACEMODE	Set interface type, Receive (Rx)/Transmit (Tx), for COM ports	interfacemode [port] rxtype txtype [responses]
4	COM	COM port configuration control	com [port] bps [parity [databits [stopbits [handshake [echo[break]]]]]]
15	CLOCKADJUST	Enable clock adjustments	clockadjust switch
18	RESET	Perform a hardware reset	reset [delay]
19	SAVECONFIG	Save current configuration in non-volatile memory	saveconfig
20	FRESET	Clear almanac model, or user configuration data, which is stored in NVM and followed by a card reset.	freset [target]
22	MODEL	Switch to a previously AUTHed model	model model
27	ASSIGN	Assign individual satellite channel to a PRN	assign channel [state] prn [Doppler [window]]
28	ASSIGNALL	Assign all satellite channels to a PRN	assignall [system] [state] prn [Doppler [window]]
29	UNASSIGN	Unassign a previously ASSIGNed channel	unassign channel
30	UNASSIGNALL	Unassign all previously ASSIGNed channels	unassignall [system]
36	UNLOG	Remove log from logging control	unlog [port] datatype
38	UNLOGALL	Remove all logs from logging control	unlogall [port]
44	FIX	Constrain to fixed height, position or velocity	fix type [param1 [param2 [param3]]]
49	AUTH	Add authorization code for new model	auth [state] part1 part2 part3 part4 part5 model [date]
50	ECUTOFF	Set satellite elevation cut-off	ecutoff angle
78	USERDATUM	Set user-customized datum	userdatum semimajor flattening dx dy dz rx ry rz scale
91	RTKELEVMASK	Set the RTK mask angle	rtkelevmask type [angle]

Message ID	Command	Description	Syntax
92	RTKSVENTRIES	Set the number of satellites to use in corrections	rtksventries number
95	STATUSCONFIG	Configure various status mask fields in RXSTATUSEVENT log	statusconfig type word mask
97	RTKCOMMAND	Reset the RTK filter or set the filter to default settings	rtkcommand action
98	ANTENNAPOWER	Control power to low-noise amplifier of an active antenna	antennapower flag
102	SETAPPROXTIME	Set an approximate GPS time	setapproxtime week sec
127	DGPSTIMEOUT	Set maximum age of differential data accepted	dgpstimeout delay
131	SETRTCM16	Enter an ASCII text message to be sent out in the RTCM data stream	SETRTCM16 text
137	LOCKOUT	Prevent the GPSCard from using a satellite by specifying its PRN	lockout prn
138	UNLOCKOUT	Reinstate a satellite in the solution computation	unlockout prn
139	UNLOCKOUTALL	Reinstate all previously locked out satellites	unlockoutall
142	DGPSEPHEMDELAY	DGPS ephemeris delay	dgpsephemdelay delay
144	DGPSTXID	DGPS transmit ID	dgpstxid type ID
160	DATUM	Choose a DATUM name type	datum datum
162	SETNAV	Set start and destination waypoints	setnav fromlat fromlon tolat tolon track offset from-point to-point
173	POSAVE	Implement position averaging for base station	posave[state] maxtime[maxhstd[maxvstd]]
177	SEND	Send an ASCII message to any of the communications ports	send port data
178	SENDHEX	Send non-printable characters in hexadecimal pairs	sendhex port length data
180	MAGVAR	Set magnetic variation correction	magvar type [correction [stddev]]
182	RTKBASELINE	Initialize RTK with a static baseline	rtkbaseline type [par1 par2 par3 [2sigma]]

Message ID	Command	Description	Syntax
183	RTKDYNAMICS	Set the RTK dynamics mode	rtkdynamics mode
184	RTKSOLUTION	Set RTK carrier phase ambiguity type (Float or Fixed) or disable	rtksolution type
197	NVMRESTORE	Restore NVM data after a failure in NVM	nvmrestore
214	UNDULATION	Choose undulation	undulation option [separation]
230	EXTERNALCLOCK	Set external clock parameters	externalclock clocktype [freq] [h0 [h1 [h2]]]
232	FREQUENCYOUT	Sets the output pulse train available on VARF.	frequencyout [switch] [pulsewidth] [period]
258	DYNAMICS	Tune receiver parameters	dynamics dynamics
269	CSMOOTH	Set carrier smoothing	csmooth L1time [L2time]
312	WAASCORRECTION	Set SBAS test mode and PRN	waascorrection keyword [prn] [mode]
377	SETAPPROXPOS	Set an approximate position	setapproxpos lat lon height
429	ADJUST1PPS	Adjust the receiver clock	adjust1pps mode [period] [offset]
430	CLOCKCALIBRATE	Adjust the control parameters of the clock steering loop	clockcalibrate mode [period] [width] [slope] [bandwidth]
431	COMCONTROL	Control the hardware control lines of the RS232 ports	comcontrol port signal control
467	ASSIGNOMNI	Set OmniSTAR satellite communication parameters	assignomni mode freq baud
493	PSRDIFFSOURCE	Set the pseudorange correction source	psrdiffsource type ID
494	RTKSOURCE	Set the RTK correction source	rtksource type ID

When the GPSCard is first powered up, or after an FRESET command, all commands will revert to the factory default settings. The SAVECONFIG command can be used to modify the power-on defaults. Use the RXCONFIG log to determine command and log settings.

✉ FRESET STANDARD causes all previously stored user configurations saved to non-volatile memory to be erased (including Saved Config, Saved Almanac, and Saved Ephemeris).

2.4 MiLLennium GPSCard Compatibility

Table 10 shows the MiLLennium commands that are comparable to current OEM4 family commands.

Table 10: OEM4 Family Command Comparison

MiLLennium Command	Comparable OEM4/EURO4 Command
ACCEPT	INTERFACEMODE
ANTENNAPOWER	ANTENNAPOWER
ASSIGN	ASSIGN
CLOCKADJUST	CLOCKADJUST
COMn	COM
COMn_DTR	COMCONTROL
COMn_RTS	COMCONTROL
CONFIG	Not currently supported.
CRESET	FRESET
CSMOOTH	CSMOOTH
DATUM	DATUM
DGPSTIMEOUT	DGPSTIMEOUT and DGPSEPHEMDELAY
DIFF_PROTOCOL	Not currently supported.
DYNAMICS	DYNAMICS
ECUTOFF	ECUTOFF
EXTERNALCLOCK	EXTERNALCLOCK
FIX HEIGHT	FIX HEIGHT
FIX POSITION	FIX POSITION
FIX VELOCITY	FIX VELOCITY
FREQUENCY_OUT	FREQUENCYOUT
FRESET	FRESET
HELP or ?	Not currently supported.
IONOMODEL	Not currently supported
LOCKOUT	LOCKOUT
LOG	LOG
MAGVAR	MAGVAR
MESSAGES	INTERFACEMODE
POSAVE	POSAVE
RESET	RESET
RESETHEALTH	Not currently supported.
RESETHEALTHALL	Not currently supported.
RINEX	Not currently supported.
RTCM16T	SETRTCM16
RTCMRULE	Not currently supported.
RTKMODE	RTKBASELINE, RTKCOMMAND, RTKDYNAMICS, RTKELEVMASK, RTKSOLUTION and RTKSVENTRIES
SAVEALMA	See the FRESET command on *Page 68*.
SAVECONFIG	SAVECONFIG
SEND	SEND
SENDHEX	SENDHEX
SETDGPSID	DGPSTXID, PSRDIFFSOURCE and RTKSOURCE
SETHEALTH	Not currently supported.
SETL1OFFSET	Not currently supported.
SETNAV	SETNAV
SETTIMESYNC	ADJUST1PPS
UNASSIGN	UNASSIGN
UNASSIGNALL	UNASSIGNALL
UNDULATION	UNDULATION
UNFIX	FIX NONE
UNLOCKOUT	UNLOCKOUT
UNLOCKOUTALL	UNLOCKOUTALL
UNLOG	UNLOG
UNLOGALL	UNLOGALL
USERDATUM	USERDATUM
VERSION	See the VERSION log on *Page 232*
WAASCORRECTION	WAASCORRECTION

2.5 Factory Defaults

When the receiver is first powered up, or after a FRESET command (see *Page 67*), all commands will revert to the factory default settings. The SAVECONFIG command (see *Page 90*) can be used to modify these defaults. Use the RXCONFIG log (see *Page 215*) to reference command and log settings.

The factory defaults are:

```
ANTENNAPOWER ON
ASSIGNOMNI USER 1551489 1200
CLOCKADJUST ENABLE
COM COM1 9600 N 8 1 N OFF ON
COM COM2 9600 N 8 1 N OFF ON
COM COM3 9600 N 8 1 N OFF ON
COMCONTROL COM1 RTS DEFAULT
COMCONTROL COM2 RTS DEFAULT
COMCONTROL COM3 RTS DEFAULT
CSMOOTH 100 5
DATUM WGS84
DGPSEPHEMDELAY 120
DGPSTIMEOUT 300
DGPSTXID AUTO “ANY”
DYNAMICS AIR
ECUTOFF 5
EXTERNALCLOCK DISABLE
FIX NONE
FREQUENCYOUT DISABLE
INTERFACEMODE COM1 NOVATEL NOVATEL ON
INTERFACEMODE COM2 NOVATEL NOVATEL ON
INTERFACEMODE COM3 NOVATEL NOVATEL ON
LOG COM1 RXSTATUSEVENTA ONNEW 0 0 HOLD
LOG COM2 RXSTATUSEVENTA ONNEW 0 0 HOLD
LOG COM3 RXSTATUSEVENTA ONNEW 0 0 HOLD
MAGVAR CORRECTION 0 0
POSAVE OFF
PSRDIFFSOURCE AUTO “ANY”
RTKCOMMAND USE_DEFAULTS
RTKSOLUTION AUTO
RTKBASELINE UNKNOWN 0 0 0 0
RTKDYNAMICS DYNAMIC
RTKELEVMASK AUTO 0
RTKSVENTRIES 12
RTKSOURCE AUTO “ANY”
SETNAV 90.0 0.0 90.0 0.0 0.0 from to
STATUSCONFIG PRIORITY STATUS 0
STATUSCONFIG PRIORITY AUX1 0x00000008
STATUSCONFIG PRIORITY AUX2 0
STATUSCONFIG SET STATUS 0x00000000
STATUSCONFIG SET AUX1 0
STATUSCONFIG SET AUX2 0
STATUSCONFIG CLEAR STATUS 0x00000000
STATUSCONFIG CLEAR AUX1 0
STATUSCONFIG CLEAR AUX2 0
UNDULATION TABLE 0
USERDATUM 6378137.0 298.2572235628 0.0 0.0 0.0 0.0 0.0 0.0 0.0
WAASCORRECTION DISABLE 0 NONE
```

2.6 Command Reference

2.6.1 ADJUST1PPS

This command is used to adjust the receiver clock. This command is typically used when the receiver is not adjusting its own clock and is using an external reference frequency. To disable the automatic adjustment of the clock, refer to the CLOCKADJUST command on *Page 46*. To configure the receiver to use an external reference oscillator, refer to the EXTERNALCLOCK command on *Page 62*. The ADJUST1PPS command can be used to:

a) Manually shift the phase of the clock

b) Adjust the phase of the clock so that the output 1PPS signal matches an external signal

c) Set the receiver clock close to that of another GPS receiver

d) Set the receiver clock exactly in phase of another GPS receiver

✉ The resolution of the clock synchronization is 50 ns.

Figure 1 shows the IPPS alignment between a Master and a Slave receiver.

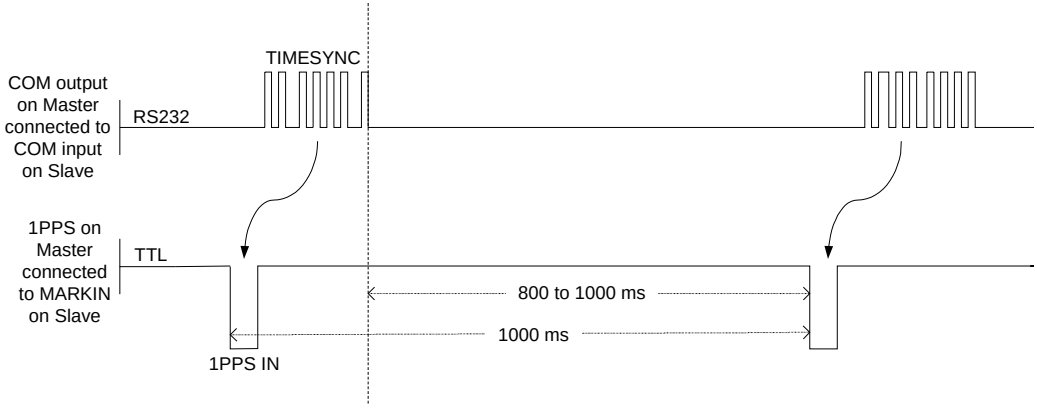


Figure 1: 1PPS Alignment

See also the TIMESYNC log on *Page 229* and the *Transfer Time Between Receivers* section in *Volume 1* of this manual.

Abbreviated ASCII Syntax: **Message ID: 429**

ADJUST1PPS mode [period][offset]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	mode	OFF	0	Disables ADJUST1PPS (default).	Enum	4	H
		MANUAL	1	Immediately shifts the receivers time by the offset field in ns. The period field has no effect in this mode. This command does not affect the clock state			
		MARK	2	Shifts the receiver time to align its 1PPS with the signal received in the MARKIN port adjusted by the offset field in ns. The effective shift range is ± 0.5 s.			
		MARKWITHTIME [a]	3	Shifts the receiver time to align its 1PPS with the signal received in the MARKIN port adjusted by the offset field in ns, and sets the receiver Time of Week (TOW) and week number, to that embedded in a received TIMESYNC log, see *Page 229*. It also sets the receiver Time Status to that embedded in the TIMESYNC log, which must have arrived between 800 and 1000 ms prior to the MARKIN event (presumably the 1PPS from the master), or it will be rejected as an invalid message.			
		TIME	4	If the receiver clock is not at least COARSE adjusted, this command enables the receiver to COARSE adjust its time upon receiving a valid TIMESYNC log in any of the COM ports. The clock state embedded in the TIMESYNC log must be at least FINE or FINESTEERING before it will be considered. The receiver does not use the MARKIN event in this mode.			

3	period	ONCE	0	The time is synchronized only once (default). The ADJUST1PPS command must be re-issued if another synchronization is required.	Enum	4	H+4
		CONTINUOUS	1	The time is continuously monitored and the receiver clock is corrected if an offset of more than 50 ns is detected.			
4	offset	-2147483648 to +2147483647		Allows the operator to shift the slave clock in 50 ns increments. In MANUAL mode, this command will apply an immediate shift of this offset in ns to the receiver clock. In MARK and MARKWITHTIME mode, this offset will shift the receiver clock with respect to the time of arrival of the MARKIN event. If this offset is zero the slave will align its 1PPS to that of the signal received in its MARKIN port. For example, if this value was set to 50, then the slave would set its 1PPS 50 ns ahead of the input signal and if this value was set to -100 then the slave would set its clock to 100 ns behind the input signal.	Long	4	H+8

a. It is presumed that the TIMESYNC log, see *Page 229*, was issued by a Master GPS receiver within 1000 ms, but not less than 800 ms, of the last 1PPS event, see *Figure 8, 1PPS Alignment-TIMESYNC Log on Page 229.* Refer also to the *Transfer Time Between Receivers* section in *Volume 1* of this manual.

ASCII Example:

ADJUST1PPS MARK CONTINUOUS 240

2.6.2 ANTENNAPOWER

This command enables or disables the supply of electrical power from the internal (see *Volume 1* of this manual for information on supplying power to the antenna) power source of the card to the low-noise amplifier (LNA) of an active antenna.

There are several bits in the Receiver Status (see *Table 70, Receiver Status on Page 223*) that pertain to the antenna. These bits indicate whether the antenna is powered (internally or externally) and whether it is open circuited or short circuited.

On startup, the ANTENNAPOWER is set to ON.

Abbreviated ASCII Syntax: **Message ID: 98**

ANTENNAPOWER flag

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	flag	OFF	0	Disables internal powering of antenna.	Enum	4	H
		ON	1	Enables internal powering of antenna.			

ASCII Example:

ANTENNAPOWER ON

2.6.3 ASSIGN

✉ The ASSIGN command should only be used by advanced users of GPS.

This command may be used to aid in the initial acquisition of a satellite by allowing you to override the automatic satellite/channel assignment and reacquisition processes with manual instructions. The command specifies that the indicated tracking channel search for a specified satellite at a specified Doppler frequency within a specified Doppler window. The instruction will remain in effect for the specified SV channel and PRN, even if the assigned satellite subsequently sets. If the satellite Doppler offset of the assigned SV channel exceeds that specified by the Window parameter of the ASSIGN command, the satellite may never be acquired or re-acquired. To cancel the effects of ASSIGN, you must issue the ASSIGN AUTO, UNASSIGN or UNASSIGNALL command. This will return SV channel control to the automatic search engine.

✉ Assigning a SV channel will set the forced assignment bit in the channel tracking status field which is reported in the RANGE and TRACKSTAT logs

✉ Assigning a PRN to a SV channel does not remove the PRN from the search space of the automatic searcher; only the SV channel is removed (i.e. The searcher may search and lock onto this PRN on another channel). The automatic searcher only searches for PRNs 1 to 32 for GPS channels and PRNs 120 to 138 for SBAS channels.

Table 11: Channel State

Binary	ASCII	Description
0	IDLE	Set the SV channel to not track any satellites
1	ACTIVE	Set the SV channel active (default)
2	AUTO	Tell the receiver to automatically assign PRN codes to channels

Abbreviated ASCII Syntax: **Message ID: 27**

ASSIGN channel [state] [prn [Doppler [window]]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively (see *1.1, Message Types on Page 12)*.	-	H	0
2	channel	0 to 11		Desired SV channel number from 0 to 11 inclusive (where channel 0 is the first SV channel and channel 11 is the last).	ULong	4	H
3	state	See *Table 11, Channel State on Page 39*		Set the SV channel state.	Enum	4	H+4
4	prn	1 to 37, 120 to 138		Optional satellite PRN code from 1-37 for GPS channels and 120-138 for SBAS channels. If not included in the command line, the state parameter must be set to IDLE.	Long	4	H+8
5	Doppler	-100 000 to 100 000 Hz		Current Doppler offset of the satellite Note: Satellite motion, receiver antenna motion and receiver clock frequency error must be included in the calculation of Doppler frequency. (default = 0)	Long	4	H+12
6	window	0 to 10 000 Hz		Error or uncertainty in the Doppler estimate above. Note: This is a ± value. Example: 500 for ± 500 Hz. (default = 4 500)	ULong	4	H+16

ASCII Example 1:

ASSIGN 0,ACTIVE,29,0,2000

In example 1, the first SV channel is acquiring satellite PRN 29 in a range from -2000 Hz to 2000 Hz until the satellite signal has been detected.

ASCII Example 2:

ASSIGN 11,28,-250,0

SV channel 11 is acquiring satellite PRN 28 at an offset of -250 Hz only.

ASCII Example 3:

ASSIGNA 11,IDLE

SV channel 11 is idled and will not attempt to search for satellites.

2.6.4 ASSIGNALL

✉ The ASSIGNALL command should only be used by advanced users of GPS.

This command allows you to override the automatic satellite/channel assignment and reacquisition processes for all GPSCard channels with manual instructions. This command works the same way as ASSIGN except that it affects all SV channels.

Abbreviated ASCII Syntax: **Message ID: 28**

ASSIGNALL [system][state][prn [Doppler [window]]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	system	See *Table 12*		System that SV channel is tracking.	Enum	4	H
3	state	See *Table 11, Channel State on Page 39*		Set the SV channel state.	Enum	4	H+4
4	prn	1 to 37, 120-138		Optional satellite PRN code from 1-37 for GPS channels and 120-138 for SBAS channels. If not included in the command line, the state parameter must be set to idle.	Long	4	H+8
5	Doppler	-100 000 to 100 000 Hz		Current Doppler offset of the satellite Note: Satellite motion, receiver antenna motion and receiver clock frequency error must be included in the calculation of Doppler frequency. (default = 0)	Long	4	H+12
6	window	0 to 10 000 Hz		Error or uncertainty in the Doppler estimate above.This is a ± value (e.g. 500 for ± 500 Hz). (default =4500)	ULong	4	H+16

Table 12: Channel System

Binary	ASCII	Description
0	GPSL1	GPS L1 dedicated SV channels only.
1	GPSL1L2	GPS L1 and L2 dedicated SV channels only.
2	NONE	No dedicated SV channels.
3	ALL	All channels (default).
4	WAASL1	SBAS SV channels only.

ASCII Example 1:

ASSIGNALL GPSL1,ACTIVE,29,0,2000

In example 1, all GPS L1 dedicated SV channels are set to active and trying to acquire PRN 29 in a range from -2000 Hz to 2000 Hz until the satellite signal has been detected.

ASCII Example 2:

ASSIGNALL GPSL1L2,28,-250,0

All L1 and L2 dedicated SV channels are trying to acquire satellite PRN 28 at -250 Hz only.

ASCII Example 3:

ASSIGNALL GPSL1,IDLE

All L1 only dedicated SV channels are idled and are not attempting to search for satellites.

2.6.5 ASSIGNOMNI

This command allows you to use manual instructions to ensure that the receiver searches for a specified OmniSTAR satellite at a specified frequency with a specified baud rate.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Abbreviated ASCII Syntax: **Message ID: 467**

ASSIGNOMNI mode freq baud

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively (see *1.1, Message Types on Page 12*).	-	H	0
2	mode	USER	0	Set the mode to USER and enter specific frequency and baud rate values.	Enum	4	H
3	freq	1525000 to 1560000		OmniSTAR service frequency of satellite (kHz).	Ulong	4	H+4
4	baud	300, 600, 1200, 2400 or 4800		Data rate for communication with OmniSTAR satellite.	Ulong	4	H+8

ASCII Example:

ASSIGNOMNI USER 1525000 4800

2.6.6 AUTH

This command is used to add or remove authorization codes from the GPSCard. Authorization codes are used to authorize models of software for a receiver. The receiver is capable of keeping track of five authorization codes at one time. The MODEL command can then be used to switch between authorized models. The VALIDMODELS log will list the current available models in the GPSCard. This simplifies the use of multiple software models on the same receiver.

If there is more than one valid model in the card, the card will either use the model of the last auth code entered via the AUTH command or the model that was selected by the MODEL command, whichever was done last. Both the AUTH and MODEL commands cause a reset automatically.

✉ Authorization codes are firmware version specific. If the GPSCard firmware is updated, it is necessary to acquire new authorization codes for the required models. If you wish to update the firmware in the receiver, please contact NovAtel Customer Service.

WARNING! **Removing an authorization code will cause the receiver to permanently lose this information.**

Abbreviated ASCII Syntax: **Message ID: 49**

AUTH [state] part1 part2 part3 part4 part5 model model [date]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	state	REMOVE	0	Remove the authcode from the system.	Enum	4	H
		ADD	1	Add the authcode to the system. (default)			
3	part1	4 digit hexadecimal (0-FFFF)		Authorization code section 1.	ULong	4	H+4
4	part2	4 digit hexadecimal (0-FFFF)		Authorization code section 2.	ULong	4	H+8
5	part3	4 digit hexadecimal (0-FFFF)		Authorization code section 3.	ULong	4	H+12
6	part4	4 digit hexadecimal (0-FFFF)		Authorization code section 4.	ULong	4	H+16
7	part5	4 digit hexadecimal (0-FFFF)		Authorization code section 5.	ULong	4	H+20
8	model	Alpha numeric	Null terminated	Model name of the receiver	String [max. 16]	16	H+24
9	date	Numeric	Null terminated	Expiry date entered as yymmdd in decimal.	String [max. 7]	8[a]	H+40

a. In the binary log case an additional 1 byte of padding is added to maintain 4 byte alignment

Input Examples:

AUTH ADD 1234 5678 9ABC DEF0 1234 OEM4L1L2 990131

AUTH 1234 5678 9ABC DEF0 1234 OEM4L1L2

2.6.7 CLOCKADJUST

All oscillators have some inherent drift. By default the receiver attempts to steer the receiver's clock to accurately match GPS time. If for some reason this is not desired, this behavior can be disabled using the CLOCKADJUST command. The TIME log can then be used to monitor clock drift.

✉ 1. The CLOCKADJUST command should only be used by advanced users of GPS.

2. If the CLOCKADJUST command is ENABLED, and the receiver is configured to use an external reference frequency (set in the EXTERNALCLOCK command, see *Page 62*, for an external clock - TCXO, OCXO, RUBIDIUM, CESIUM, or USER), then the clock steering process will take over the VARF output pins and may conflict with a previously entered FREQUENCYOUT command, see *Page 67*.
3. When disabled, the range measurement bias errors will continue to accumulate with clock drift.
4. Pseudorange, carrier phase and Doppler measurements may jump if the CLOCKADJUST mode is altered while the receiver is tracking.
5. When disabled, the time reported on all logs may be offset from GPS time. The 1PPS output may also be offset. The amount of this offset may be determined from the TIME log, see *Page 227*.
6. A discussion on GPS time may be found in *Section 1.3, GPS Time Status on Page 20.*

Abbreviated ASCII Syntax: **Message ID: 15**

CLOCKADJUST switch

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	switch	DISABLE	0	Disallow adjustment of internal clock.	Enum	4	H
		ENABLE	1	Allow adjustment of internal clock.			

ASCII Example:

CLOCKADJUST DISABLE

2.6.8 CLOCKCALIBRATE

This command is used to adjust the control parameters of the clock steering loop. The receiver must be enabled for clock steering before these values can take effect. Refer to the CLOCKAJUST command, see *Page 46*, to enable or disable this feature. The receiver by default steers its INTERNAL VCTCXO but can be commanded to control an EXTERNAL reference oscillator. Use the EXTERNALCLOCK command, see *Page 62*, to configure the receiver to use an external reference oscillator. If the receiver is configured for an external reference oscillator and configured to adjust its clock, then the clock steering loop will attempt to steer the external reference oscillator through the use of the VARF signal. Note that the clock steering control process will conflict with the manual FREQUENCYOUT command, see *Page 67*. It is expected that the VARF signal is used to provide a stable reference voltage by the use of a filtered charge pump type circuit (not supplied).

To disable the clock steering process, issue the CLOCKADJUST DISABLE command.

The current values used by the clock steering process are listed in the CLOCKSTEERING log, see *Page 130*.

Abbreviated ASCII Syntax: **Message ID: 430**

CLOCKCALIBRATE mode [period] [width] [slope] [bandwidth]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	mode	SET	0	Sets the period, pulsewidth, slope, and bandwidth values into NVM for the currently selected steered oscillator (INTERNAL or EXTERNAL).	Enum	4	H
		AUTO	1	Forces the receiver to do a clock steering calibration to measure the slope (change in clock drift rate with a 1 bit change in pulse width), and required pulsewidth, to zero the clock drift rate. After the calibration, these values along with the period and bandwidth are entered into NVM and will then be used from this point forward on the selected oscillator.			
		OFF	2	Terminates a calibration process currently underway.			
3	period	0 to 262144		Signal period in 25 ns steps. Frequency Output = 40,000,000 / Period. (default = 0)	Ulong	4	H+4

4	pulsewidth	The valid range for this parameter is 10% to 90% of the period.	Sets the initial pulse width that should provide a near zero drift rate from the selected oscillator being steered. The valid range for this parameter is 10% to 90% of the period. The default value is 2000. If this value is not known, (in the case of a new external oscillator) then it should be set to ½ the period and the mode should be set to AUTO to force a calibration.	Ulong	4	H+8
5	slope		This value should correspond to how much the clock drift will change with a 1 bit change in the pulsewidth m/s/bit. The default values for the slope used for the INTERNAL and EXTERNAL clocks is -2.0 and -0.01 respectively. If this value is not known, then its value should be set to 1.0 and the mode should be set to AUTO to force a calibration. Once the calibration process is complete and using a slope value of 1.0, the receiver should be recalibrated using the measured slope and pulsewidth values (Fields #6 and #4 of the CLOCKSTEERING log, see *Page 130*). This process should be repeated until the measured slope value remains constant (less than a 5% change).	Float	4	H+12
6	bandwidth		This is the value used to control the smoothness of the clock steering process. Smaller values will result in slower and smoother changes to the receiver clock. Larger values will result in faster responses to changes in oscillator frequency and faster startup clock pull-in. The default values are 0.03 and 0.001 Hz respectively for the INTERNAL and EXTERNAL clocks.	Float	4	H+16

ASCII Example:

CLOCKCALIBRATE AUTO

2.6.9 COM

This command permits you to configure the receiver's asynchronous serial port communications drivers.

The current COM port configuration can be reset to its default state at any time by sending it two hardware break signals of 250 milliseconds each, spaced by fifteen hundred milliseconds (1.5 seconds) with a pause of at least 250 milliseconds following the second break. This will:

- Stop the logging of data on the current port (see the UNLOGALL command on *Page 101)*
- Clear the transmit and receive buffers on the current port
- Return the current port to its default settings (see *Page 34* for details)
- Set the interface mode to NovAtel for both input and output (see the INTERFACEMODE command on Page 69)

See also *Section 2.5, Factory Defaults on Page 34* for a description of the factory defaults, and the COMCONFIG log on *Page 139*.

✉ The COMCONTROL command, see *Page 51*, may conflict with handshaking of the selected COM port. If handshaking is enabled, then unexpected results may occur.

Abbreviated ASCII Syntax: **Message ID: 4**

COM [port] bps [parity[databits[stopbits[handshake[echo[break]]]]]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	See *Table 13, Serial Port Identifiers, on Page 50*		Port to configure. (default = THISPORT)	Enum	4	H
3	bps/baud	300, 600, 900, 1200, 2400, 9600, 19200, 38400, 57600, 115200, or 230400		Communication baud rate (bps).	ULong	4	H+4
4	parity	See *Table 14, Parity, on Page 50*		Parity	Enum	4	H+8
5	databits	7 or 8		Number of data bits (default = 8).	ULong	4	H+12
6	stopbits	1 or 2		Number of stop bits (default = 1).	ULong	4	H+16
7	handshake	See *Table 15, Handshaking on Page 50*		Handshaking	Enum	4	H+20
8	echo	OFF	0	No echo (default).	Enum	4	H+24
		ON	1	Transmit any input characters as they are received.			
9	break	OFF	0	Disable break detection	Enum	4	H+28
		ON	1	Enable break detection (default)			

ASCII Example:

COM COM1,57600,N,8,1,N,OFF,ON

Table 13: Serial Port Identifiers

ASCII	Binary	Description
COM1	1	COM port 1
COM2	2	COM port 2
COM3	3	COM port 3
THISPORT	6	The current COM port
ALL	8	All COM ports

Table 14: Parity

Binary	ASCII	Description
0	N	No parity (default)
1	E	Even parity
2	O	Odd parity

Table 15: Handshaking

Binary	ASCII	Description
0	N	No handshaking (default)
1	XON	XON/XOFF software handshaking
2	CTS	CTS/RTS hardware handshaking

2.6.10 COMCONTROL

This command is used to control the hardware control lines of the RS232 ports. The TOGGLEPPS mode of this command is typically used to supply a timing signal to a host PC computer. The accuracy of controlling the COM control signals is better than 900 µs. The other modes are typically used to control custom peripheral devices.

✉ This command may conflict with handshaking of the selected COM port. If handshaking is enabled, then unexpected results may occur.

Abbreviated ASCII Syntax: **Message ID: 431**

COMCONTROL port signal control

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	COM1	1	RS232 port to control. Valid ports are COM1, COM2 and COM3.	Enum	4	H
		COM2	2				
		COM3	3				
3	signal	RTS	0	COM signal to control. The controllable COM signals are RTS and DTR. The DTR signal is only applicable to COM2.	Enum	4	H+4
		DTR	1				
4	control	DEFAULT	0	Disables this command and returns the COM signal to its default state.	Enum	4	H+8
		FORCEHIGH	1	Immediately forces the signal high.			
		FORCELOW	2	Immediately forces the signal low.			
		TOGGLE	3	Immediately toggles the current sate of the signal.			
		TOGGLEPPS	4	Toggles the state of the selected signal within 900 µs after each 1PPS event. The state change of the signal will lag the 1PPS by an average value of 450 µs. The delay of each pulse will vary by a uniformly random amount less than 900 µs.			

ASCII Example:

COM COM1 9600 N 8 1 N

COMCONTROL COM1 RTS FORCELOW

2.6.11 CSMOOTH

This command sets the amount of carrier smoothing to be performed on the code measurements. An input value of 100 corresponds to approximately 100 seconds of smoothing. Upon issuing the command, the locktime for all tracking satellites is reset to zero. From this point each code smoothing filter is restarted. The user must wait for at least the length of smoothing time for the new smoothing constant to take full effect. The optimum setting for this command is dependent on your application.

Abbreviated ASCII Syntax: **Message ID: 269**

CSMOOTH L1time [L2time]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	L1time	2-2000		L1 carrier smoothing time constant, in seconds.	Ulong	4	H
3	[L2time]	2-2000		L2 carrier smoothing time constant, in seconds. Default = 5.	Ulong	4	H+4

Abbreviated ASCII Example:

CSMOOTH 500

✉ **The csmooth command should only be used by advanced GPS users.** The shorter the carrier smoothing the more noise there will be. If you are at all unsure please call NovAtel Customer Service Department, see the *Customer Service* section at the start of *Volume 1* of this manual.

✉ It may not be suitable for every GPS application. When using CSMOOTH in differential mode, the same setting should be used at both the base and rover station, if both the base and rover stations are using the same type of receiver (OEM3 or OEM4). However if the base and rover stations use different types of receivers (OEM3 and OEM4), it is recommended that the default CSMOOTH value is used at each receiver.

2.6.12 DATUM

This command permits you to select the geodetic datum for operation of the receiver. If not set, the factory default value is WGS84. See *Table 17* below for a complete listing of all available predefined datums. See the USERDATUM command for user definable datums. The datum you select will cause all position solutions to be based on that datum.

Abbreviated ASCII Syntax: **Message ID: 160**

DATUM datum

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	datum	See *Table 17, Datum Transformation Parameters, on Page 54*		User defined datum with parameters specified by the USERDATUM command	Enum	4	H

ASCII Example:

DATUM WGS84

The following tables contain the internal ellipsoid parameters and transformation parameters used in the GPSCard. The values contained in these tables were derived from the following DMA technical reports:

1. TR 8350.2 Department of Defence World Geodetic System 1984 and Relationships with Local Geodetic Systems - Revised March 1, 1988.
2. TR 8350.2B Supplement to Department of Defence World Geodetic System 1984 Technical Report - Part II - Parameters, Formulas, and Graphics for the Practical Application of WGS84 - December 1, 1987.

Table 16: Reference Ellipsoid Constants

ELLIPSOID	ID CODE	a (meters)	1/f	f
Airy 1830	AW	6377563.396	299.3249647	0.00334085064038
Modified Airy	AM	6377340.189	299.3249647	0.00334085064038
Australian National	AN	6378160.0	298.25	0.00335289186924
Bessel 1841	BR	6377397.155	299.1528128	0.00334277318217
Clarke 1866	CC	6378206.4	294.9786982	0.00339007530409
Clarke 1880	CD	6378249.145	293.465	0.00340756137870
Everest (India 1830)	EA	6377276.345	300.8017	0.00332444929666
Everest (Brunei & E.Malaysia)	EB	6377298.556	300.8017	0.00332444929666
Everest (W.Malaysia & Singapore)	ED	6377304.063	300.8017	0.00332444929666
Geodetic Reference System 1980	RF	6378137.0	298.257222101	0.00335281068118
Helmert 1906	HE	6378200.0	298.30	0.00335232986926
Hough 1960	HO	6378270.0	297.00	0.00336700336700
International 1924	IN	6378388.0	297.00	0.00336700336700
South American 1969	SA	6378160.0	298.25	0.00335289186924
World Geodetic System 1972	WD	6378135.0	298.26	0.00335277945417
World Geodetic System 1984	WE	6378137.0	298.257223563	0.00335281066475

Table 17: Datum Transformation Parameters

Datum ID #	NAME	DX	DY	DZ	DATUM DESCRIPTION	ELLIPSOID
1	ADIND	-162	-12	206	Adindan (Ethiopia, Mali, Senegal & Sudan)	Clarke 1880
2	ARC50	-143	-90	-294	ARC 1950 (SW & SE Africa)	Clarke 1880
3	ARC60	-160	-8	-300	ARC 1960 (Kenya, Tanzania)	Clarke 1880
4	AGD66	-133	-48	148	Australian Geodetic Datum 1966	Australian National
5	AGD84	-134	-48	149	Australian Geodetic Datum 1984	Australian National
6	BUKIT	-384	664	-48	Bukit Rimpah (Indonesia)	Bessel 1841
7	ASTRO	-104	-129	239	Camp Area Astro (Antarctica)	International 1924
8	CHATM	175	-38	113	Chatum 1971 (New Zealand)	International 1924
9	CARTH	-263	6	431	Carthage (Tunisia)	Clarke 1880
10	CAPE	-136	-108	-292	CAPE (South Africa)	Clarke 1880
11	DJAKA	-377	681	-50	Djakarta (Indonesia)	Bessel 1841
12	EGYPT	-130	110	-13	Old Egyptian	Helmert 1906
13	ED50	-87	-98	-121	European 1950	International 1924
14	ED79	-86	-98	-119	European 1979	International 1924
15	GUNSG	-403	684	41	G. Segara (Kalimantan - Indonesia)	Bessel 1841
16	GEO49	84	-22	209	Geodetic Datum 1949 (New Zealand)	International 1924
17	GRB36	375	-111	431	Great Britain 1936 (Ordinance Survey)	Airy 1830
18	GUAM	-100	-248	259	Guam 1963 (Guam Island)	Clarke 1866
19	HAWAII	89	-279	-183	Hawaiian Hawaii (Old)	International 1924
20	KAUAI	45	-290	-172	Hawaiian Kauai (Old)	International 1924
21	MAUI	65	-290	-190	Hawaiian Maui (Old)	International 1924
22	OAHU	56	-284	-181	Hawaiian Oahu (Old)	International 1924
23	HERAT	-333	-222	114	Herat North (Afghanistan)	International 1924
24	HJORS	-73	46	-86	Hjorsey 1955 (Iceland)	International 1924
25	HONGK	-156	-271	-189	Hong Kong 1963	International 1924

Table 17: Datum Transformation Parameters

26	HUTZU	-634	-549	-201	Hu-Tzu-Shan (Taiwan)	International 1924
27	INDIA	289	734	257	Indian (India, Nepal, Bangladesh)	Everest (EA)
28	IRE65	506	-122	611	Ireland 1965	Modified Airy
29	KERTA	-11	851	5	Kertau 1948 (West Malaysia and Singapore)	Everest (ED)
30	KANDA	-97	787	86	Kandawala (Sri Lanka)	Everest (EA)
31	LIBER	-90	40	88	Liberia 1964	Clarke 1880
32	LUZON	-133	-77	-51	Luzon (Philippines excluding Mindanoa Is.)	Clarke 1866
33	MINDA	-133	-70	-72	Mindanoa Island	Clarke 1866
34	MERCH	31	146	47	Merchich (Morocco)	Clarke 1880
35	NAHR	-231	-196	482	Nahrwan (Saudi Arabia)	Clarke 1880
36	NAD83	0	0	0	N. American 1983 (Includes Areas 37-42)	GRS-80
37	CANADA	-10	158	187	N. American Canada 1927	Clarke 1866
38	ALASKA	-5	135	172	N. American Alaska 1927	Clarke 1866
39	NAD27	-8	160	176	N. American Conus 1927	Clarke 1866
40	CARIBB	-7	152	178	N. American Caribbean	Clarke 1866
41	MEXICO	-12	130	190	N. American Mexico	Clarke 1866
42	CAMER	0	125	194	N. American Central America	Clarke 1866
43	MINNA	-92	-93	122	Nigeria (Minna)	Clarke 1880
44	OMAN	-346	-1	224	Oman	Clarke 1880
45	PUERTO	11	72	-101	Puerto Rica and Virgin Islands	Clarke 1866
46	QORNO	164	138	-189	Qornoq (South Greenland)	International 1924
47	ROME	-255	-65	9	Rome 1940 Sardinia Island	International 1924
48	CHUA	-134	229	-29	South American Chua Astro (Paraguay)	International 1924
49	SAM56	-288	175	-376	South American (Provisional 1956)	International 1924
50	SAM69	-57	1	-41	South American 1969	S. American 1969
51	CAMPO	-148	136	90	S. American Campo Inchauspe (Argentina)	International 1924
52	SACOR	-206	172	-6	South American Corrego Alegre (Brazil)	International 1924
53	YACAR	-155	171	37	South American Yacare (Uruguay)	International 1924
54	TANAN	-189	-242	-91	Tananarive Observatory 1925 (Madagascar)	International 1924
55	TIMBA	-689	691	-46	Timbalai (Brunei and East Malaysia) 1948	Everest (EB)
56	TOKYO	-128	481	664	Tokyo (Japan, Korea and Okinawa)	Bessel 1841
57	TRIST	-632	438	-609	Tristan Astro 1968 (Tristan du Cunha)	International 1924
58	VITI	51	391	-36	Viti Levu 1916 (Fiji Islands)	Clarke 1880
59	WAK60	101	52	-39	Wake-Eniwetok (Marshall Islands)	Hough 1960
60	WGS72	0	0	4.5	World Geodetic System - 72	WGS72
61	WGS84	0	0	0	World Geodetic System - 84	WGS84
62	ZANDE	-265	120	-358	Zanderidj (Surinam)	International 1924
63	USER	0	0	0	User Defined Datum Defaults	User *

Notes:
* Default user datum is WGS84.
* Also see the *USERDATUM* command on *Page 102*.
* These GPSCard logs report Datum used according to the "GPSCard Datum ID" column: BESTPOS, PPOS, and MATCHEDPOS.

2.6.13 DGPSEPHEMDELAY *DGPS*

Abbreviated ASCII Syntax: **Message ID: 142**

DGPSEPHEMDELAY delay

The DGPSEPHEMDELAY command is used to set the ephemeris delay when operating as a base station. The ephemeris delay sets a time value by which the base station will continue to use the old ephemeris data. A delay of 120 to 300 seconds will typically ensure that the rover stations have collected updated ephemeris. After the delay period is passed, the base station will begin using new ephemeris data.

The factory default of 120 seconds matches the RTCM standard.

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	delay	0 to 600 s		Minimum time delay before new ephemeris is used.	ULong	4	H

ASCII Example (reference):

DGPSEPHEMDELAY 120

✉ The RTCA Standard stipulates that a base station shall wait five minutes after receiving a new ephemeris before transmitting differential corrections to rover stations that are using the RTCA standard. This time interval ensures that the rover stations will have received the new ephemeris, and will compute differential positioning based upon the same ephemeris. Therefore, for RTCA base stations, the recommended ephemeris delay is 300 seconds.

2.6.14 DGPSRXID

This command is now obsolete and has been replaced by the PSRDIFFSOURCE and RTKSOURCE commands. Please see *Pages 79* and *88* respectively for more information on these commands.

2.6.15 DGPSTIMEOUT *DGPS*

Abbreviated ASCII Syntax: **Message ID: 127**

DGPSTIMEOUT delay

This command is used to set the maximum age of pseudorange differential data that will be used when operating as a rover station. Pseudorange differential data received that is older than the specified time will be ignored. RTK differential data is fixed at 60 seconds and cannot be changed. See DGPSEPHEMDELAY on *Page 56* to set the ephemeris changeover delay for base stations.

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	delay	2 to 1000 s		Maximum pseudorange differential age.	ULong	4	H

ASCII Example (rover):

DGPSTIMEOUT 60

✉ The RTCA Standard for SCAT-I stipulates that the maximum age of differential correction messages cannot be greater than 22 seconds. Therefore, for RTCA rover users, the recommended DGPS delay setting is 22.

2.6.16 DGPSTXID DGPS

Abbreviated ASCII Syntax: **Message ID: 144**

DGPSTXID type ID

This command sets the station ID value for the receiver when it is transmitting corrections. This allows for the easy identification of which base station was the source of the data.

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 27, DGPS Type, on Page 80*		ID Type	Enum	4	H
3	ID	String [max. 5] or "ANY"		ID string ANY type defaults: RTCM - 0 RTCA - AAAA CMR - 0 The following range values are in affect: $0 \leq$ CMR ID ≤ 31 $0 \leq$ RTCM ID ≤ 1023 RTCA: any four character string containing only alpha (a-z) or numerical characters (0-9)	String [max. 5]	8[a]	H+4

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

ASCII Examples:

DGPSTXID RTCM 2 - using a specific type and ID

DGPSTXID CMR 30 - using a specific type and ID

DGPSTXID CMR "ANY" - using the default CMR ID

2.6.17 DYNAMICS

This command adjusts the receiver dynamics to that of your environment. It is used to optimally tune receiver parameters.

Abbreviated ASCII Syntax: **Message ID: 258**

DYNAMICS dynamics

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	dynamics	See *Table 18*		Receiver dynamics based on the user's.	Enum	4	H

Table 18: User Dynamics

Binary	ASCII	Description
0	AIR	Receiver is in an aircraft
1	LAND	Receiver is in a land vehicle with velocity less than 110 km/h (30 m/s)
2	FOOT	Receiver is being carried by a person with velocity less than 11 km/h (3 m/s)

Example:

DYNAMICS FOOT

2.6.18 ECUTOFF

This command sets the elevation cut-off angle for tracked satellites. The GPSCard will not start automatically searching for a satellite until it rises above the cut-off angle. Tracked satellites that fall below the cut-off angle will no longer be tracked unless they were manually assigned (see the ASSIGN command).

In either case, satellites below the ECUTOFF angle will be eliminated from the internal position and clock offset solution computations.

This command permits a negative cut-off angle; it could be used in these situations:

- the antenna is at a high altitude, and thus can look below the local horizon
- satellites are visible below the horizon due to atmospheric refraction

Abbreviated ASCII Syntax: **Message ID: 50**

ECUTOFF angle

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	angle	±90.0 degrees		Elevation cut-off angle relative to horizon.	Float	4	H

ASCII Example:

ECUTOFF 10.0

✉ 1. Care must be taken when using ECUTOFF because the signals from lower elevation satellites are travelling through more atmosphere and are therefore degraded. Use of satellites below 5 degrees is not recommended.

2. This command does not affect the RTK mode elevation cut-off angle. It only affects which satellites are tracked. See the RTKELEVMASK command on *Page 86*.

3. This command does not affect the tracking of SBAS satellites.

2.6.19 EXTERNALCLOCK

Overview

The EXTERNALCLOCK command allows the Euro4 to operate with an optional external oscillator. The user is able to optimally adjust the clock model parameters of the Euro4 for various types of external clocks.

✉ 1. This command will affect the interpretation of the CLOCKMODEL log.

2. If the EXTERNALCLOCK command is enabled and set for an external clock (TCXO, OCXO, RUBIDIUM, CESIUM, or USER) and the CLOCKADJUST command, see *Page 46*, is ENABLED, then the clock steering process will take over the VARF output pins and may conflict with a previously entered FREQUENCYOUT command, see *Page 62*.

There are two steps involved in using an external oscillator:

1. Follow the procedure outlined in your Euro4 installation/operation manual (*Volume 1* of this manual) for connecting an external oscillator to your Euro4.
2. Using the EXTERNALCLOCK command, select a standard oscillator and its operating frequency.

Theory

An unsteered oscillator can be approximated by a three-state clock model, with two states representing the range bias and range bias rate, and a third state assumed to be a Gauss-Markov (GM) process representing the range bias error generated from satellite clock dither. The third state is included because the Kalman filter assumes an (unmodeled) white input error. The significant correlated errors produced by SA clock dither are obviously not white and the Markov process is an attempt to handle this kind of short-term variation.

The internal units of the new clock model's three states (offset, drift and GM state) are meters, meters per second, and meters. When scaled to time units for the output log, these become seconds, seconds per second, and seconds, respectively. Note that the old units of the third clock state (drift rate) were meters per second per second.

The user has control over 3 process noise elements of the linear portion of the clock model. These are the h_0, h_{-1}, and h_{-2} elements of the power law spectral density model used to describe the frequency noise characteristics of oscillators:

$$S_y(f) = \frac{h_{-2}}{f^2} + \frac{h_{-1}}{f} + h_0 + h_1 f + h_2 f^2$$

where f is the sampling frequency and $S_y(f)$ is the clock's power spectrum. Typically only h_0, h_{-1}, and h_{-2} affect the clock's Allan variance and the clock model's process noise elements.

Usage

Before using an optional external oscillator, several clock model parameters must be set. There are default settings for a voltage-controlled temperature-compensated crystal oscillator (VCTCXO), ovenized crystal oscillator (OCXO), Rubidium and Cesium standard; or, the user may choose to supply customized settings.

Abbreviated ASCII Syntax: **Message ID: 230**

EXTERNALCLOCK clocktype [freq] [h0 [h1 [h2]]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	clocktype	See *Table 19 on Page 64*		Clock type	Enum	4	H
3	freq	5MHz	0	Frequency	Enum	4	H+4
		10MHz	1				
4	h0	See *Table 20 on Page 64*		Timing standards	Double	8	H+8
5	h1				Double	8	H+16
6	h2				Double	8	H+24

ASCII Example:

EXTERNALCLOCK DISABLE

or

EXTERNALCLOCK USER 10MHZ 1.0167e-23 6.87621e-25 8.1762e-26

or

EXTERNALCLOCK TCXO 5MHZ 1.000000000e-21
1.000000000e-20 1.000000000e-20

Table 19: Clock Type

ASCII	Binary	Description
DISABLE	0	Turns the external clock input off, reverts back to the on-board VCTCXO
TCXO	1	Sets the defaults for a VCTCXO
OCXO	2	Sets the defaults for an OCXO
RUBIDIUM	3	Sets the defaults for a rubidium oscillator
CESIUM	4	Sets the defaults for a cesium oscillator
USER	5	Defines custom process noise elements

Table 20: Default Values of Process Noise Elements

Timing Standard	h_0	h_{-1}	h_{-2}
VCTCXO	1.0 e-21	1.0 e-20	2.0 e-20
OCXO	2.51 e-26	2.51 e-23	2.51 e-22
rubidium	1.0 e-23	1.0 e-22	1.3 e-26
cesium	2.0 e-20	7.0 e-23	4.0 e-29
user (min. / max)	1.0 e-31 $\leq h_0 \leq$ 1.0 e-18	1.0 e-31 $\leq h_{-1} \leq$ 1.0 e-18	1.0 e-31 $\leq h_{-2} \leq$ 1.0 e-18

2.6.20 FIX

This command fixes various parameters of the GPSCard such as height, position, and velocity. For various applications, fixing these values can assist in improving acquisition times and accuracy of position or corrections. For example, fixing the position and height is a requirement for differential base stations as it provides a truth position to base the differential corrections from.

Abbreviated ASCII Syntax: **Message ID: 44**

FIX type [param1 [param2 [param3]]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 22 on Page 66*		Fix type	Enum	4	H
3	param1	See *Table 21*		Parameter 1.	Double	8	H + 4
4	param2			Parameter 2.	Double	8	H + 12
5	param3			Parameter 3.	Double	8	H + 20

ASCII Example:

FIX HEIGHT 4.567

✉ Any setting other than FIX POSITION will disable output of differential corrections.

Table 21: FIX Parameters

ASCII Type Name	Parameter 1	Parameter 2	Parameter 3
AUTO	Not used	Not used	Not used
HEIGHT	Height (-1000 to 20000000 m)	Not used	Not used
NONE	Not used	Not used	Not used
POSITION	Lat (-90 to 90 deg)	Lon (-360 to 360 deg)	Hgt (-1000 to 20000000 m)[a]
VELOCITY	VX (-999.9 to 999.9 m/s)	VY (-999.9 to 999.9 m/s)	VZ (-999.9 to 999.9 m/s)

a. For a discussion on Geoidal Height, refer to *Volume 1, Appendix C, Height Relationships*.

Table 22: FIX Types

ASCII Name	Binary Value	Description
NONE	0	Unfix. Clears any previous FIX commands.
AUTO	1	Configures the GPSCard to fix the height at the last calculated value if the number of satellites available is insufficient for a 3-D solution. This provides a 2-D solution. Height calculation will resume when the number of satellites available allows a 3-D solution.
HEIGHT	2	Configures the GPSCard in 2-D mode with its height constrained to a given value. The command would be used mainly in marine applications where height in relation to mean sea level may be considered to be approximately constant. The height entered using this command is always referenced to the geoid (mean sea level, see the BESTPOS log on *Page 122*) and uses units of meters. The receiver is capable of receiving and applying differential corrections from a base station while FIX HEIGHT is in effect. The FIX HEIGHT command will override any previous FIX HEIGHT or FIX POSITION command Note: This command only affects pseudorange corrections and solutions, and so has no meaning within the context of RT-2 and RT-20.
POSITION	3	Configures the GPSCard with its position fixed. This command is used when it is necessary to generate differential corrections. For both pseudorange and differential corrections, this command must be properly initialized before the GPSCard can operate as a GPS base station. Once initialized, the receiver will compute differential corrections for each satellite being tracked. The computed differential corrections can then be output to rover stations by utilizing any of the following GPSCard differential corrections data log formats: RTCM, RTCA, or CMR. See the *OEM4 User Manual Volume 1* for information on using the receiver for differential applications. The values entered into the FIX POSITION command should reflect the precise position of the base station antenna phase centre. Any errors in the FIX POSITION coordinates will directly bias the corrections calculated by the base receiver. The GPSCard performs all internal computations based on WGS84 and the datum command is defaulted as such. The datum in which you choose to operate (by changing the DATUM command) will be internally converted to and from WGS84. Therefore, all differential corrections are based on WGS84, regardless of your operating datum. The FIX POSITION command will override any previous FIX HEIGHT or FIX POSITION command settings.
VELOCITY	4	Configures the GPSCard with its velocity fixed. It accepts ECEF XYZ velocity values in units of meters per second (m/s). This information is only used by the tracking loops of the receiver to aid in reacquisition of satellites after loss of lock, otherwise it is ignored. It is not used in the position solution and velocity calculations. This command is only useful for very high dynamics where expected velocity changes during the signal blockage of more than 100 meters per second can occur. The FIX VELOCITY command will override any previous FIX HEIGHT or FIX POSITION command.

2.6.21 FREQUENCYOUT

This command sets the output pulse train available on the variable frequency (VARF) pin.

The output waveform will be coherent with the 1PPS output.

✉ If the CLOCKADJUST command is ENABLED, see *Page 46*, and the receiver is configured to use an external reference frequency (set in the EXTERNALCLOCK command, see *Page 62*, for an external clock - TCXO, OCXO, RUBIDIUM, CESIUM, or USER), then the clock steering process will take over the VARF output pins and may conflict with a previously entered FREQUENCYOUT command.

Abbreviated ASCII Syntax: **Message ID: 232**

FREQUENCYOUT [switch] [pulsewidth] [period]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	switch	DISABLE	0	Disable will cause the output to be fixed low (default)	Enum	4	H
		ENABLE	1	Enables customized frequency output			
3	pulse width	(0 to 262144)		Number of 25 ns steps for which the output will be high. Duty cycle = pulsewidth / period. Must be less than or equal to the period. (default = 0). If pulsewidth is the same as the period, the output will be a high DC signal. If pulsewidth is 1/2 the period, then the output will be a square wave.	Ulong	4	H+4
4	period	(0 to 262144)		Signal period in 25 ns steps. Frequency Output = 40,000,000 / Period. (default = 0).	Ulong	4	H+8

ASCII Example:

FREQUENCYOUT ENABLE 2 4

This example will generate a 50% duty cycle 10 MHz square wave.

2.6.22 FRESET

This command clears data which is stored in non-volatile memory. Such data includes the almanac, ephemeris, and any user-specific configurations. The commands, ephemeris and almanac can be cleared by using the STANDARD target. The model can only be cleared by using the MODEL target. The receiver is forced to hardware reset.

Abbreviated ASCII Syntax: **Message ID: 20**

FRESET [target]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	target	See *Table 23*		What data is to be reset by the receiver.	Enum	4	H

Input Example:

FRESET COMMAND

✉ FRESET STANDARD (which is also the default) causes all previously stored configurations that were saved to non-volatile memory to be erased (includes Saved Config, Saved Ephemeris and Saved Almanac).

Table 23: FRESET Target

Binary	ASCII	Description
0	STANDARD	Resets commands, ephemeris and almanac (default)
1	COMMAND	Resets the stored commands (saved configuration)
2	GPSALMANAC	Resets the stored almanac
3	GPSEPHEM	Resets stored ephemeris
5	MODEL	Resets the currently selected model

2.6.23 INTERFACEMODE DGPS

This command allows the user to specify what type of data a particular port on the GPSCard can transmit and receive. The receive type tells the receiver what type of data to accept on the specified port. The transmit type tells the receiver what kind of data it can generate. For example, you would set the receive type on a port to RTCA in order to accept RTCA differential corrections.

It is also possible to disable or enable the generation or transmission of command responses for a particular port. Disabling of responses is important for applications where data is required in a specific form and the introduction of extra bytes may cause problems, for example RTCA, RTCM or CMR. Disabling a port prompt is also useful when the port is connected to a modem or other device that will respond with data the receiver does not recognize.

When INTERFACEMODE *port* NONE NONE OFF is set, the specified port will be disabled from interpreting any input or output data. Therefore, no commands or differential corrections will be decoded by the specified port. Data can be passed through the disabled port and be output from an alternative port using the pass-through logs PASSCOM1, PASSCOM2 and PASSCOM3. See *Page 170* for details on the PASSCOM logs and the *Operation* chapter, in Volume 1 of this manual, for information on pass-through logging. See also the COMCONFIG log on *Page 139*.

Abbreviated ASCII Syntax: **Message ID: 3**

INTERFACEMODE [port] rxtype txtype [responses]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	See *Table 13, Serial Port Identifiers, on Page 50*		Serial port identifier (default = THISPORT)	Enum	4	H
3	rxtype	See *Table 24, below*		Receive interface mode	Enum	4	H+4
4	txtype			Transmit interface mode	Enum	4	H+8
5	responses	OFF	0	Turn response generation off	Enum	4	H+12
		ON	1	Turn response generation on (default)			

ASCII Example:

INTERFACEMODE COM1 RTCA NOVATEL ON

Table 24: Serial Port Interface Modes

ASCII Mode Name	Binary Value	Description
NONE	0	The port accepts/generates nothing
NOVATEL	1	The port accepts/generates NovAtel commands and logs
RTCM	2	The port accepts/generates RTCM corrections
RTCA	3	The port accepts/generates RTCA corrections
CMR	4	The port accepts/generates CMR corrections

2.6.24 LOCKOUT

This command will prevent the GPSCard from using a satellite by de-weighting its range in the solution computations. Note that the LOCKOUT command does not prevent the GPSCard from tracking an undesirable satellite. This command must be repeated for each satellite to be locked out.

See also the UNLOCKOUT and UNLOCKOUTALL commands.

Abbreviated ASCII Syntax: **Message ID: 137**

LOCKOUT prn

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	prn	1-37		A single satellite PRN number to be locked out.	Ulong	4	H

Input Example:

LOCKOUT 8

2.6.25 LOG

Many different types of data can be logged using several different methods of triggering the log events. Every log element can be directed to any combination of the three COM ports (and virtual ports which are described in *Volume 1* of this manual). The ONTIME trigger option requires the addition of the *period* parameter. See *Chapter 3, Data Logs on Page 104* for further information and a complete list of data log structures.

Table 25 shows the binary command format while *Table 26* shows the ASCII command format.

The optional parameter [hold] will prevent a log from being removed when the UNLOGALL command is issued. To remove a log which was invoked using the [hold] parameter requires the specific use of the UNLOG command.

The [port] parameter is optional. If [port] is not specified, [port] is defaulted to the port that the command was received on.

✉ The OEM4 family of receivers can handle 20 logs at a time. If you attempt to log more than 20 logs at a time, the receiver will respond with an Insufficient Resources error.

✉ Maximum flexibility for logging data is provided to the user by these logs. The user is cautioned, however, to recognize that each log requested requires additional CPU time and memory buffer space. Too many logs may result in lost data and degraded CPU performance. Receiver overload can be monitored using the idle-time field and buffer overload bits of the Receiver Status in any log header.

✉ Using the ‘onmark’ trigger with a MARKTIME or MARKPOS log, is no different than using the ‘onnew’ trigger.

✉ Only the MARKPOS log, the MARKTIME log, and ‘polled’ log types are generated ‘on the fly’ at the exact time of the mark. Synchronous and asynchronous logs will output the most recently available data.

Abbreviated ASCII Syntax: **Message ID: 1**

LOG [port] message [trigger [period [offset [hold]]]]

Table 25: LOG Command Binary Format

Field	Field Name	Binary Value	Description	Field Type	Binary Bytes	Binary Offset
1	header	(See *Table 4, Binary Message Header Structure, on Page 17*)	This field contains the message header.	-	H	0
2	port	See *Table 5, Serial Port Identifiers on Page 18*	Output port.	Enum	4	H
3	message	Any valid message ID	Message ID of log to output.	UShort	2	H+4
4	message type	Bits 0-4 = Reserved Bits 5-6 = Format 00 = Binary 01 = ASCII 10 = Abbreviated ASCII, NMEA 11 = Reserved Bit 7 = Response Bit (see *Section 1.2, Responses on Page 19*) 0 = Original Message 1 = Response Message	Message type of log.	Char	1	H+6
5	reserved			Char	1	H+7
6	trigger	0 = ONNEW	Output when the message is updated (not necessarily changed).	Enum	4	H+8
		1 = ONCHANGED	Output when the message is changed.			
		2 = ONTIME	Output on a time interval.			
		3 = ONNEXT	Output only the next message.			
		4 = ONCE	Output only the current message.			
		5 = ONMARK	Output when a MARKIN electrical event is detected. [a]			
7	period		Log period (for ONTIME trigger) in seconds	Double	8	H+12
8	offset		Offset for period (ONTIME trigger) in seconds. If you wished to log data at 1 second after every minute you would set the period to 60 and the offset to 1	Double	8	H+20
9	hold	0 = NOHOLD	Allow log to be removed by the UNLOGALL command.	Enum	4	H+28
		1 = HOLD	Prevent log from being removed by the UNLOGALL command.			

a. This happens on the external MARKIN pin, refer to the *Technical Specifications* appendix in *Volume 1* of this manual for more details on pinouts.

Table 26: LOG Command ASCII Format

Field	Field Name	ASCII Value	Description	Field Type
1	header	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII or ASCII respectively.	-
2	port	See *Table 5, Serial Port Identifiers on Page 18*	Output port (default = THISPORT)	Enum
3	message	Any valid message name, optionally with an A or B suffix.	Message name of log to output.	Char [ ]
4	trigger	ONNEW	Output when the message is updated (not necessarily changed).	Enum
		ONCHANGED	Output when the message is changed.	
		ONTIME	Output on a time interval.	
		ONNEXT	Output only the next message.	
		ONCE	Output only the current message. (default)	
		ONMARK	Output when a MARKIN electrical event is detected. [a]	
5	period		Log period (for ONTIME trigger) in seconds (default = 0)	Double
6	offset		Offset for period (ONTIME trigger) in seconds. If you wished to log data at 1 second after every minute you would set the period to 60 and the offset to 1 (default = 0)	Double
7	hold	NOHOLD	Allow log to be removed by the UNLOGALL command. (default)	Enum
		HOLD	Prevent log from being removed by the UNLOGALL command.	

a. This happens on the external MARKIN pin, refer to the *Technical Specifications* appendix in *Volume 1* of this manual for more details on pinouts.

Abbreviated ASCII Example 1:

LOG COM1 BESTPOSA ONTIME 7 2.5 HOLD

The above example shows the BESTPOSA log is logging to COM port 1 at 7 second intervals and offset by 2.5 seconds (output at 2.5, 9.5, 16.5 seconds and so on). The [hold] parameter is set so that logging is not disrupted by the UNLOGALL command.

To send a log only one time, the trigger option can be ignored.

Abbreviated ASCII Example 2:

LOG COM1 BESTPOSA ONCE 0.000000 0.000000 NOHOLD

See *Section 2.1, Command Formats on Page 22* for additional examples.

2.6.26 MAGVAR

The GPSCard computes directions referenced to True North. Use this command (magnetic variation correction) if you intend to navigate in agreement with magnetic compass bearings. The correction value entered here will cause the "bearing" field of the NAVIGATE log to report bearing in degrees Magnetic. The GPSCard will compute the magnetic variation correction if you use the auto option. See *Figure 2, Illustration of Magnetic Variation & Correction on Page 75.*

The receiver calculates values of magnetic variation for given values of latitude, longitude and time using the International Geomagnetic Reference Field (IGRF) 95 spherical harmonic coefficients and IGRF time corrections to the harmonic coefficients.

Abbreviated ASCII Syntax: **Message ID: 180**

MAGVAR type [correction] [std dev]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	AUTO	0	Use IGRF corrections	Enum	4	H
		CORRECTION	1	Use the correction supplied			
3	correction	± 180.0 degrees		Magnitude of correction **(Required field if type = Correction)**	Float	4	H+4
4	std_dev	± 180.0 degrees		Standard deviation of correction (default = 0)	Float	4	H+8

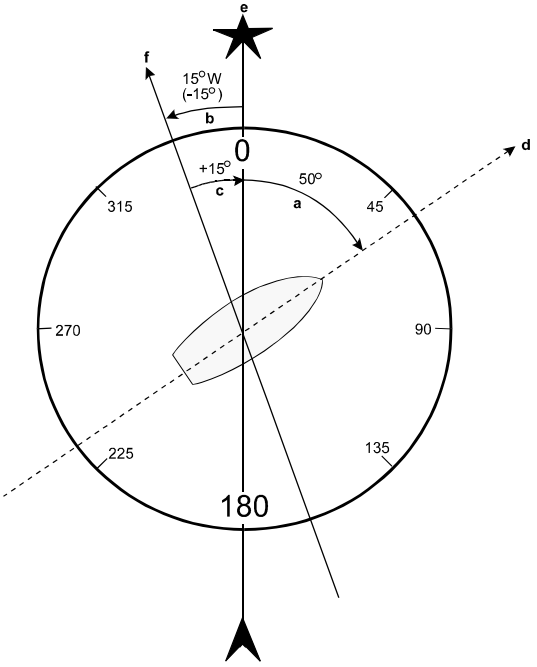
ASCII Example 1:

MAGVAR AUTO 0.000000000 0.000000000

ASCII Example 2:

MAGVAR CORRECTION 15.000000000 0.000000000

Figure 2: Illustration of Magnetic Variation & Correction



Reference	Description
a	True Bearing
b	Local Magnetic Variation
c	Local Magnetic Variation Correction (inverse of magnetic variation)
a + c	Magnetic Bearing
d	Heading: 50° True, 60° Magnetic
e	True North
f	Local Magnetic North

2.6.27 MODEL

This command is used to switch the card between models previously added with the AUTH command. When this command is issued, the GPSCard will save this model as the active model. The active model will now be used on every subsequent startup. The MODEL command causes an automatic reset.

Use the VALIDMODELS log to output a list of available models for your card. The VALIDMODELS log is described on *Page 232*. Use the VERSION log to output the active model, see *Page 233*.

Abbreviated ASCII Syntax: **Message ID: 22**

MODEL model

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	model	Max 16 character null-terminated string (including the null)		Model name.	String [max. 16]	16	H

Input Example:

MODEL RT2W

✉ If you switch to an expired model, the receiver will reset and enter into an error state. You will need to switch to a valid model to continue.

2.6.28 NVMRESTORE

This command restores non-volatile memory (NVM) data after a NVM Fail error. This failure is indicated by bit 13 of the receiver error word being set (see also *RXSTATUS, Page 220* and *RXSTATUSEVENT, Page 225*). If corrupt NVM data is detected, the receiver will remain in the error state and continue to flash an error code on the Status LED until the NVMRESTORE command is issued (see the chapter on *Built-In Status Tests* in *Volume 1* of this manual for further explanation).

The possibility of NVM failure is extremely remote, however, if it should occur it is likely only a small part of the data is corrupt. This command is used to remove the corrupt data and restore the receiver to an operational state. The data lost could be the user configuration, almanac, model, or other reserved information.

If you have more than one auth-code and the saved model is lost then the model may need to be entered using the MODEL command or it will be automatically saved in NVM on the next startup. If the almanac was lost, a new almanac will automatically be saved when the next complete almanac is received (after approximately 15 minutes of continuous tracking). If the user configuration was lost it will have to be re-entered by the user. This could include communication port settings.

✉ The factory default is 9600, n, 8, 1

After entering the NVMRESTORE command and resetting the card, the communications link may have to be re-established at a different baud rate from the previous connection.

Abbreviated ASCII Syntax: **Message ID: 197**

NVMRESTORE

2.6.29 POSAVE

This command implements position averaging for base stations. Position averaging will continue for a specified number of hours or until the estimated averaged position error is within specified accuracy limits. Averaging will stop when the time limit or the horizontal standard deviation limit or the vertical standard deviation limit is achieved. When averaging is complete, the FIX POSITION command will automatically be invoked.

If the maximum time is set to 1 hour or larger, positions will be averaged every 10 minutes and the standard deviations reported in the AVEPOS log should be correct if SA is on. If the maximum time is set to less than 1 hour, positions will be averaged once per minute and the standard deviations reported in the log will likely not be accurate; therefore, the optional horizontal and vertical standard deviation limits cannot be used.

If you initiate differential logging, then issue the POSAVE command followed by the SAVECONFIG command the GPSCard will average positions after every power-on or reset, and will then invoke the FIX POSITION command to enable it to send differential corrections.

Abbreviated ASCII Syntax: **Message ID: 173**

POSAVE [state] maxtime [maxhstd [maxvstd]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	state	ON	1	Enable or disable position averaging. (default = ON)	Enum	4	H
		OFF	0				
3	maxtime	0.01 - 100 hours		Maximum amount of time that positions are to be averaged. Only becomes optional if State = OFF.	Float	4	H+4
4	maxhstd	0 - 100 m		Desired horizontal standard deviation. (default = 0)	Float	4	H+8
5	maxvstd	0 - 100 m		Desired vertical standard deviation. (default = 0)	Float	4	H+12

ASCII Example:

POSAVE 24 1 2

2.6.30 PSRDIFFSOURCE DGPS

This command sets the station ID value which identifies the base station from which to accept RTCA1, RTCM1 or OmniSTAR Virtual Base Station (VBS) differential corrections. This is useful when the GPSCard is receiving corrections from multiple base stations. See also the RTKSOURCE command on *Page 88*.

✉ 1. When a valid PSRDIFFSOURCE command is received, the current correction is removed immediately rather than waiting for the time specified in DGPSTIMEOUT, see *Page 57*.

2. In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Abbreviated ASCII Syntax: **Message ID: 493**

PSRDIFFSOURCE type ID

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 27*		ID Type. All types may revert to WAAS (if enabled) or SINGLE position types. See also *Table 39, Position or Velocity Type, on Page 123*.	Enum	4	H
3	ID	Char [5] or ANY		ID string	Char[5]	8[a]	H+4

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

ASCII Examples:

1. Select only SBAS:
   RTKSOURCE NONE
   PSRDIFFSOURCE NONE
   WAASCORRECTION ENABLE 0 WAASTESTMODE
2. Enable OmniSTAR HP and VBS:
   RTKSOURCE OMNISTAR
   PSRDIFFSOURCE OMNISTAR
3. Enable RTK and PSRDIFF from RTCM, with a fall-back to SBAS:
   RTKSOURCE RTCM ANY
   PSRDIFFSOURCE RTCM ANY
   WAASCORRECTION ENABLE 0 WAASTESTMODE

Table 27: DGPS Type

Binary	ASCII	Description
0	RTCM[a]	RTCM ID: 0 ≤ RTCM ID ≤ 1023 **or** ANY
1	RTCA[a]	RTCA ID: Any four character string containing only alpha (a-z) or numerical characters (0-9) **or** ANY
2	CMR[ab]	CMR ID: 0 ≤ CMR ID ≤ 31 **or** ANY
3	OMNISTAR[c]	In the PSRDIFFSOURCE command, OMNISTAR enables OmniSTAR VBS and disables other DGPS types. OmniSTAR VBS produces RTCM-type corrections. In the RTKSOURCE command, OMNISTAR enables OmniSTAR HP (if allowed) and disables other RTK types. OmniSTAR HP has its own filter, computes corrections in RTK float mode or within about 10 cm accuracy.
10	AUTO[c]	In the PSRDIFFSOURCE command, AUTO means the first received RTCM or RTCA message has preference over an OmniSTAR VBS message. In the RTKSOURCE command, for RTK models, AUTO means the NovAtel RTK filter is enabled and the first received RTCM, RTCA or CMR message is selected. For non-RTK models, AUTO means the OmniSTAR HP (if allowed) message is enabled.
11	NONE[c]	Disables all the DGPS and OMNISTAR types

a. Disables OmniSTAR Virtual Base Stations (VBS).
b. Available only with the RTKSOURCE command, see *Page 88.*
c. ID parameter is ignored

2.6.31 RESET

This command performs a hardware reset. Following a RESET command, the GPSCard will initiate a cold-start bootup. Therefore, the receiver configuration will revert either to the factory default if no user configuration was saved or the last SAVECONFIG settings.

The optional delay field is used to set the number of seconds the receiver is to wait before resetting.

Abbreviated ASCII Syntax: **Message ID: 18**

RESET [delay]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	delay			Seconds to wait before resetting. (default = 0)	Ulong	4	H

See also the FRESET and SAVECONFIG commands.

2.6.32 RTKBASELINE *RTK*

✉ This command only affects RT-2 operation and not RT-20.

This command is used in differential RTK mode to set the initial baseline information for the base station and rover station. Setting the initial baseline speeds up ambiguity resolution by indicating to the RT-2 software the exact length of the vector between the rover and base station antennas. It only affects the operation of an RT-2 system on baselines not exceeding 30 km. There are two methods of entering the baseline information: LLH and ECEF.

The first method is to use absolute LAT/LON/HEIGHT coordinates. *LAT* (in degrees) requires a decimal fraction format; a negative sign for South latitude. *LON* (in degrees) requires a decimal fraction format; a negative sign for West longitude. *HEIGHT* (in meters) can refer either to mean sea level (default) or to an ellipsoid. The optional *2*σ defines the accuracy (2 sigma, 3 dimensional) of the input position, in meters; it must be 0.03 m or less to cause the RT-2 algorithms to undergo a forced initialization to fixed integer ambiguities. If no value is entered, a default value of 0.30 m is assumed; this will not cause an initialization to occur. The optional *M* or *E* in the type field refers to the height: if “M” the height will be assumed to be above mean sea level (MSL) and if “E” the height will be ellipsoidal. Note that when an MSL height is entered, it will be converted to ellipsoidal height using the NovAtel internal undulation table or the last value entered with the “UNDULATION” command.

The other method is to use the relative ECEF vector. The Δ*X*,Δ*Y*,Δ*Z* values (in meters) represent the rover station’s position minus the base position, along each axis (in meters). The optional *2*σ defines the accuracy (2 sigma, 3 dimensional) of the input baseline, in meters; it must be 0.03 m or less to cause the RT-2 algorithms to do a forced initialization to fixed integer ambiguities. If no value is entered, a default value of 0.30 m is assumed; this will not cause an initialization to occur.

Table 28: Baseline Parameters

Type	Parameter 1	Parameter 2	Parameter 3
Unknown	N/A	N/A	N/A
LLM	Lat	Lon	MSL Ht
LLE	Lat	Lon	Ellipsoidal Ht
ECEF	ΔX	ΔY	ΔZ

Abbreviated ASCII Syntax: **Message ID: 182**

RTKBASELINE type [par1 par2 par3 [2sigma]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 29*		Set baseline type	Enum	4	H
3	par1	The baseline parameters are dependant on the type, see *Table 28, Baseline Parameters, on Page 82* (They are required when the type is not UNKNOWN)			Double	8	H+4
4	par2				Double	8	H+12
5	par3				Double	8	H+20
6	2Sigma			Accuracy (2 sigma, 3 dimensional) in meters (default = 0.3 m)	Float	4	H+28

ASCII Example:

RTKBASELINE ECEF_BASELINE 7.54 3.28 2.02 0.25

Table 29: Baseline Type

ASCII	Binary	Description
UNKNOWN	0	Unknown baseline (default).
LLM_POSITION	1	Set base to lat/lon/height with MSL height
LLE_POSITION	2	Set base to lat/lon/height with Ellipsoidal height
ECEF_BASELINE	3	Set base to ECEF

2.6.33 RTKCOMMAND *RTK*

This command provides the ability to reset the RTK filter and clear any set RTK parameters. The RESET parameter causes the RTK algorithm (RT-20 or RT-2, whichever is active) to undergo a complete reset, forcing the system to restart the ambiguity resolution calculations. The USE_DEFAULTS command executes the following commands:

RTKBASELINE UNKNOWN

RTKDYNAMICS DYNAMIC

RTKELEVMASK AUTO

RTKSOLUTION FLOAT (RT-20)

RTKSOLUTION FIXED (RT-2)

RTKSVENTRIES 12

✉ See the descriptions for the above commands in the following pages.

Abbreviated ASCII Syntax: **Message ID: 97**

RTKCOMMAND action

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	USE_DEFAULTS	0	Reset to defaults	Enum	4	H
		RESET	1	Reset RTK algorithm			

ASCII Example:

RTKCOMMAND USE_DEFAULTS

2.6.34 RTKDYNAMICS RTK

This command provides the ability to specify how the GPSCard looks at the data. There are three modes: STATIC, DYNAMIC, and AUTO. The STATIC mode forces the RTK software to treat the rover station as though it were stationary, regardless of the output of the motion detector.

✉ For reliable performance the antenna should not move more than 1-2 cm when in static mode.

DYNAMIC forces the software to treat the receiver as though it were in motion. If the receiver is undergoing very slow steady motion (<2.5 cm/sec for more than 5 seconds), you should use DYNAMIC mode (as opposed to AUTO) to prevent inaccurate results and possible resets.

On startup, the GPSCard defaults to the DYNAMIC setting.

Abbreviated ASCII Syntax: **Message ID: 183**

RTKDYNAMICS mode

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	mode	See *Table 30*		Set the dynamics mode.	Enum	4	H

ASCII Example:

RTKDYNAMICS DYNAMIC

Table 30: Dynamics Mode

ASCII	Binary	Description
AUTO	0	Automatically determine dynamics mode.
STATIC	1	Static mode.
DYNAMIC	2	Dynamic mode.

2.6.35 RTKELEVMASK *RTK*

This command sets the mask angle below which satellites will not be included. On a base station, this limits the satellites that will be transmitted in RTK observations. On a rover receiver, satellites below the mask angle will be deweighted in RTK computations.

On startup, the GPSCard defaults to the AUTO setting.

Abbreviated ASCII Syntax: **Message ID: 91**

RTKELEVMASK type [angle]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	AUTO	0	Base - all available satellites are included Rover - elevation mask varies depending on baseline length	Enum	4	H
		USER	1	User entered angle			
3	angle	0 - 90 degrees		Elevation mask angle (required for USER setting)	Float	4	H+4

ASCII Example:

RTKELEVMASK USER 10.5

2.6.36 RTKSOLUTION RTK

This command instructs the receiver as to what type of carrier phase ambiguity (fixed, float or disable) to use. There are four settings: AUTO, L1_FLOAT, L1L2_FLOAT and DISABLE. AUTO tells the receiver to use the best ambiguity type available. For RT-2, the receiver will fix the ambiguities to discrete values whenever it can safely and reliably do so. L1_FLOAT and L1L2_FLOAT tell the receiver to only use floating point ambiguities. L2 data is required for L1L2_FLOAT. DISABLE tells the receiver to turn off RTK processing.

Abbreviated ASCII Syntax: **Message ID: 184**

RTKSOLUTION type

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 31*		Carrier phase ambiguities type	Enum	4	H

ASCII Example:

RTKSOLUTION L1_FLOAT

Table 31: Type of Carrier Phase Ambiguity

ASCII	Binary	Description
AUTO	0	Fixed integer ambiguities for RT-2. L1_FLOAT for RT-20.
L1_FLOAT	1	L1 only floating point ambiguities.
L1L2_FLOAT	2	Dual frequency floating point ambiguities (reverts to L1_FLOAT for RT-20).
DISABLE	3	Disable RTK solutions.

2.6.37 RTKSOURCE RTK

This command sets the station ID value which identifies the base station from which to accept RTK (RTCM, RTCA, CMR and OmniSTAR High Performance (HP)),differential corrections. This is useful when the GPSCard is receiving corrections from multiple base stations. See also the PSRDIFFSOURCE command on *Page 79*.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Abbreviated ASCII Syntax: **Message ID: 494**

RTKSOURCE type ID

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 27, DGPS Type, on Page 80*		ID Type	Enum	4	H
3	ID	Char [5] or ANY		ID string	Char[5]	8[a]	H+4

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

ASCII Examples:

1. Select only SBAS:
   RTKSOURCE NONE
   PSRDIFFSOURCE NONE
   WAASCORRECTION ENABLE 0 WAASTESTMODE
2. Enable OmniSTAR HP and VBS:
   RTKSOURCE OMNISTAR
   PSRDIFFSOURCE OMNISTAR
3. Enable RTK and PSRDIFF from RTCM, with a fall-back to SBAS:
   RTKSOURCE RTCM ANY
   PSRDIFFSOURCE RTCM ANY
   WAASCORRECTION ENABLE 0 WAASTESTMODE

2.6.38 RTKSVENTRIES *RTK*

This command sets the number of satellites (at the highest elevation) that will be transmitted in the RTK corrections from a base station receiver. This is useful when the amount of bandwidth available for transmitting corrections is limited.

Abbreviated ASCII Syntax: **Message ID: 92**

RTKSVENTRIES number

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	number	4-12		The number of SVs to use in the solution (default = 12)	ULong	4	H

ASCII Example:

RTKSVENTRIES 7

2.6.39 SAVECONFIG

This command saves the user's present configuration in non-volatile memory. The configuration includes the current log settings, FIX settings, port configurations, etc. and is output in the RXCONFIG log, see *Page 214*. See also the FRESET command, *Page 67*.

Abbreviated ASCII Syntax: **Message ID: 19**

SAVECONFIG

✉ **Caution!:** Be careful when using this command from NovAtel's GPSolution software as the log commands used for the various windows and the COM port configuration used by GPSolution will be saved as well. This will result in unnecessary data being logged.

2.6.40 SEND

This command is used to send ASCII printable data from COM1, COM2, or COM3 to a specified communications port. This is a one-time command, therefore the data message must be preceded by the SEND command and followed by <CR> each time you wish to send data. If the data string contains delimiters (i.e. spaces, commas, tabs, etc.), the entire string must be contained within double quotation marks. Carriage return and line feed characters (e.g. hex 0D, 0A) will be appended to the sent ASCII data.

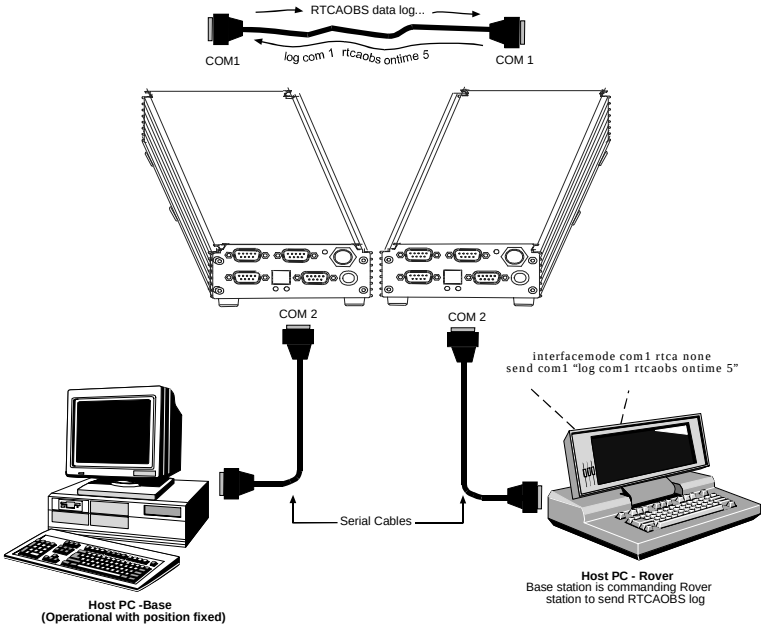
Abbreviated ASCII Syntax: **Message ID: 177**

SEND port data

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	See *Table 5, Serial Port Identifiers, on Page 18*		Output port	Enum	4	H
3	message	Max 100 character string		ASCII data to send.	String [max. 100]	100	H+4

Scenario: Assume that you are operating GPSCards as base and rover stations. It could also be assumed that the base station is unattended but operational and you wish to control it from the rover station. From the rover station, you could establish the data link and command the base station GPSCard to send differential corrections.

Figure 3: Using the SEND Command



2.6.41 SENDHEX

This command is like the SEND command except that it is used to send non-printable characters expressed as hexadecimal pairs. Carriage return and line feed characters (e.g. hex 0D, 0A) will **not** be appended to the sent data and so must be explicitly added to the data if needed.

Abbreviated ASCII Syntax: **Message ID: 178**

SENDHEX port length data

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	See *Table 5, Serial Port Identifiers on Page 18*		Output port	Enum	4	H
3	length	0 - 700		Number of hex pairs	ULong	4	H+4
4	message	limited to a 700 maximum string (1400 pair hex) by command interpreter buffer • even number of ASCII characters from set of 0-9, A-F • no spaces are allowed between pairs of characters		Data	String [max. 700]	700	H+8

Input Example:

sendhex COM1 6 143Ab5910D0A

2.6.42 SETAPPROXPOS

This command sets an approximate latitude, longitude, and height in the GPSCard. Estimating these parameters, when used in conjunction with an approximate time (see the SETAPPROXTIME command on *Page 94*), can improve satellite acquisition times and time to first fix. For more information, please refer to the *TTFF and Satellite Acquisition* appendix in *Volume 1* of this manual.

The horizontal position entered should be within 200 km of the actual receiver position. The approximate height is not critical and can normally be entered as zero. If the receiver cannot calculate a valid position within 2.5 minutes of entering an approximate position, the approximate position will be ignored.

The approximate position will not be visible in any position logs. It can be seen by entering a LOG SETAPPROXPOS message. See also the SATVIS log on *Page 226.*

Abbreviated ASCII Syntax: **Message ID: 377**

SETAPPROXPOS lat lon height

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	Lat	± 90 degrees		Approximate latitude	Double	8	H
3	Lon	± 360 degrees		Approximate longitude	Double	8	H+8
4	Height	-1000 to +20000000 m		Approximate geoidal height	Double	8	H+16

Input Example:

```
setapproxpos 51.116 -114.038 0
```

2.6.43 SETAPPROXTIME

This command sets an approximate time in the GPS card. The card will use this time as a system time until a GPS coarse time can be acquired. This can be used in conjunction with an approximate position (see the SETAPPROXPOS command on *Page 93*) to improve time to first fix. For more information, please refer to the *TTFF and Satellite Acquisition* appendix in *Volume 1* of this manual.

The time entered should be within 10 minutes of the actual GPS time.

If the week number entered does not match the broadcast week number, the receiver will reset.

See also the SATVIS log on *Page 226*.

Abbreviated ASCII Syntax: **Message ID: 102**

SETAPPROXTIME week sec

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	week	0-9999		GPS week number	Ulong	4	H
3	sec	0-604801		Number of seconds into GPS week	Double	8	H+4

Input Example:

setapproxtime 1105 425384

2.6.44 SETNAV

This command permits entry of one set of navigation waypoints (see *Figure 4 on Page 95*). The origin (FROM) and destination (TO) waypoint coordinates entered are considered on the ellipsoidal surface of the current datum (default WGS84). Once SETNAV has been set, you can monitor the navigation calculations and progress by observing the NAVIGATE log messages.

Track offset is the perpendicular distance from the great circle line drawn between the FROM lat-lon and TO lat-lon waypoints. It establishes the desired navigation path, or track, that runs parallel to the great circle line, which now becomes the offset track, and is set by entering the track offset value in meters. A negative track offset value indicates that the offset track is to the left of the great circle line track. A positive track offset value (no sign required) indicates the offset track is to the right of the great circle line track (looking from origin to destination). See *Figure 4 on Page 95* for clarification.

Abbreviated ASCII Syntax: **Message ID: 162**

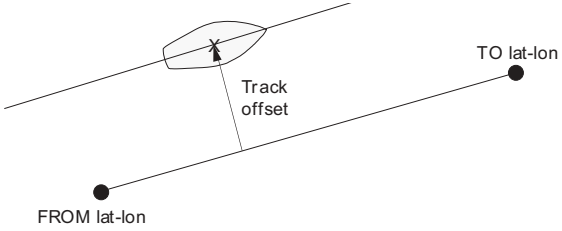
SETNAV fromlat fromlon tolat tolon track offset from-point to-point

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	fromlat	± 90 degrees		Origin latitude in units of degrees/decimal degrees. A negative sign for South latitude. No sign for North latitude.	Double	8	H
3	fromlon	± 360 degrees		Origin longitude in units of degrees/decimal degrees. A negative sign for West longitude. No sign for East longitude.	Double	8	H+8
4	tolat	± 90 degrees		Destination latitude in units of degrees/decimal degrees.	Double	8	H+16
5	tolon	± 360 degrees		Destination longitude in units of degrees/decimal degrees.	Double	8	H+24
6	track offset	± 1000 km		Waypoint great circle line offset (in kilometers); establishes offset track; positive indicates right of great circle line; negative indicates left of great circle line.	Double	8	H+32
7	from-point	6 characters max.		ASCII station name	String [max. 6]	6	H+40
8	to-point	6 characters max.		ASCII station name	String [max. 6]	6	H+46

ASCII Example:

SETNAV 51.1516 -114.16263 51.16263 -114.1516 -125.23 FROM TO

Figure 4: Illustration of SETNAV Parameters



2.6.45 SETRTCM16 DGPS & RTK

The RTCM type 16 message allows ASCII text to be transferred from a GPS base station to rover GPS receivers. The SETRTCM16 command is used to define the ASCII text at the base station. The text defined by the SETRTCM16 command can be verified in the RXCONFIG log. Once the ASCII text is defined it can be broadcast periodically by the base station with the command "log port RTCM16 ONTIME interval". The received ASCII text can be displayed at the rover by logging RTCM16T.

This command will limit the input message length to a maximum of 90 ASCII characters. If the message string contains any delimiters (i.e. spaces, commas, tabs, etc.) the entire string must be contained in double quotation marks.

Abbreviated ASCII Syntax: **Message ID: 131**

SETRTCM16 text

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	text	Max 90 character string		The text string	String [max. 90]	92[a]	H

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment

Input Example:

SETRTCM16 “Base station will shut down in 1 hour”

2.6.46 STATUSCONFIG

This command is used to configure the various status mask fields in the RXSTATUSEVENT log. These masks allow you to modify whether various status fields generate errors or event messages when they are set or cleared.

The receiver gives the user the ability to determine the importance of the status bits. In the case of the Receiver Status, setting a bit in the priority mask will cause the condition to trigger an error. This will cause the GPSCard to idle all channels, set the ERROR strobe line, flash an error code on the status LED, turn off the antenna (LNA power), and disable the RF hardware, the same as if a bit in the Receiver Error word is set. Setting a bit in an Auxiliary Status priority mask will cause that condition to set the bit in the Receiver Status word corresponding to that Auxiliary Status.

Receiver Errors automatically generate event messages. These event messages are output in RXSTATUSEVENT logs. It is also possible to have status conditions trigger event messages to be generated by the OEM4.This is done by setting/clearing the appropriate bits in the event set/clear masks. The set mask tells the GPSCard to generate an event message when the bit becomes set. Likewise, the clear mask causes messages to be generated when a bit is cleared. If you wish to disable all these messages without changing the bits, simply UNLOG the RXSTATUSEVENT logs on the appropriate ports. See also the *Built in Status Tests* chapter in *Volume 1* of this manual.

Abbreviated ASCII Syntax: **Message ID: 95**

STATUSCONFIG type word mask

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	type	See *Table 32*		Type of mask to replace	Enum	4	H
3	word	STATUS	1	Receiver Status word	Enum	4	H+4
		AUX1	2	Auxiliary 1 Status word			
4	mask	8 digit hexadecimal		The hexadecimal bit mask	Ulong	4	H+8

ASCII Example:

STATUSCONFIG SET STATUS 0028A51D

Table 32: Mask Types

ASCII	Binary	Description
PRIORITY	0	Replace the Priority mask
SET	1	Replace the Set mask
CLEAR	2	Replace the Clear mask

2.6.47 UNASSIGN

This command cancels a previously issued ASSIGN command and the SV channel reverts to automatic control (the same as ASSIGN AUTO).

Abbreviated ASCII Syntax: **Message ID: 29**

UNASSIGN channel

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	channel	0-11		Reset SV channel to automatic search and acquisition mode	ULong	4	H
3	state	See *Table 11, Channel State on Page 39*		Set the SV channel state.	Enum	4	H+4

Input Example:

unassign 11

2.6.48 UNASSIGNALL

This command cancels all previously issued ASSIGN commands for all SV channels (same as ASSIGNALL AUTO). Tracking and control for each SV channel reverts to automatic mode. See ASSIGN AUTO for more details.

Abbreviated ASCII Syntax: **Message ID: 30**

UNASSIGNALL [system]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	system	See *Table 12, Channel System, on Page 42*		System that the SV channel is tracking.	Enum	4	H

Input Example:

unassignall GPSL1

2.6.49 UNDULATION

This command permits you to either enter a specific geoidal undulation value or use the internal table of geoidal undulations. The separation values only refer to the separation between the WGS84 ellipsoid and the geoid, regardless of the datum chosen; see the BESTPOS log on *Page 122*.

Abbreviated ASCII Syntax: **Message ID: 214**

UNDULATION option [separation]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	option	TABLE	0	Use the internal undulation table value.	Enum	4	H
		USER	1	Use the user specified undulation value.			
3	separation	± 1000.0 m		The undulation value (required for the USER option).	Float	4	H+4

ASCII Example 1:

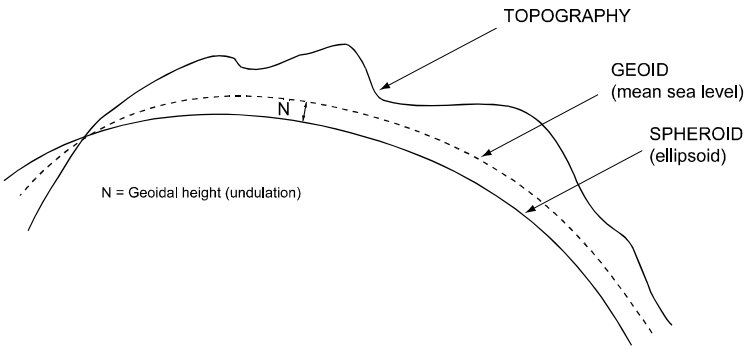
UNDULATION TABLE 0.000000000

ASCII Example 2:

UNDULATION USER -5.599999905

Please see the GPS Overview appendix in *Volume 1* of this manual for a description of the relationships in *Figure 5, Illustration of Undulation*.

Figure 5: Illustration of Undulation



2.6.50 UNLOCKOUT

This command allows a satellite which has been previously locked out (LOCKOUT command) to be reinstated in the solution computation. If more than one satellite is to be reinstated, this command must be reissued for each satellite reinstatement.

Abbreviated ASCII Syntax: **Message ID: 138**

UNLOCKOUT prn

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	prn	1-37		A single satellite PRN number to be reinstated.	Ulong	4	H

Input Example:

unlockout 8

2.6.51 UNLOCKOUTALL

This command allows all satellites which have been previously locked out (LOCKOUT command) to be reinstated in the solution computation.

Abbreviated ASCII Syntax: **Message ID: 139**

UNLOCKOUTALL

Input Example:

unlockoutall

2.6.52 UNLOG

This command permits you to remove a specific log request from the system.

The [*port*] parameter is optional. If [*port*] is not specified, it is defaulted to the port on which the command was received. This feature eliminates the need for you to know which port you are communicating on if you want logs to be removed on the same port on which you are sending this command.

Abbreviated ASCII Syntax: **Message ID: 36**

UNLOG [port] datatype

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	See *Table 5, Serial Port Identifiers on Page 18*		Port to which log is being sent. (default = THISPORT)	Enum	4	H
3	datatype	Message name/ Message ID		Message name/Message ID of log to be disabled.	ULong	4	H+4

Input Example:

unlog com1 bestposa
unlog bestposa

2.6.53 UNLOGALL

If [*port*] is specified this command disables all logs on the specified port only. All other ports are unaffected. If [*port*] is not specified this command defaults to the ALLPORTS setting.

Abbreviated ASCII Syntax: **Message ID: 38**

UNLOGALL [port]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	port	See *Table 5, Serial Port Identifiers on Page 18*		Port to clear. (default = ALLPORTS)	Enum	4	H
3	Reserved, binary value always = 0				Enum	4	H+4

Input Example:

unlogall com2_15

✉ This command does not disable logs that have the HOLD attribute (see description for LOG command). To disable logs with the HOLD attribute, use the UNLOG command.

2.6.54 USERDATUM

This command permits entry of customized ellipsoidal datum parameters. This command is used in conjunction with the DATUM command. The default setting is WGS84.

Abbreviated ASCII Syntax: **Message ID: 78**

USERDATUM semimajor flattening dx dy dz rx ry rz scale

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	semimajor	6300000.0 - 6400000.0 m		Datum Semi-major Axis (a) in meters	Double	8	H
3	flattening	290.0 - 305.0		Reciprocal Flattening, 1/f = a/(a-b)	Double	8	H+8
4	dx	± 2000.0		Datum offsets from WGS84. These will be the translation values between the user datum and WGS84 (internal reference)	Double	8	H+16
5	dy	± 2000.0			Double	8	H+24
6	dz	± 2000.0			Double	8	H+32
7	rx	± 10.0 radians		Datum Rotation Angle about X, Y and Z axis. These values will be the rotation from WGS84 to your datum. A positive sign is for clockwise rotation and a negative sign is for counter clockwise rotation.	Double	8	H+40
8	ry	± 10.0 radians			Double	8	H+48
9	rz	± 10.0 radians			Double	8	H+56
10	scale	± 10.0 ppm		Scale value is the difference in ppm between the user datum and WGS84	Double	8	H+64

ASCII Example:

USERDATUM 6378206.400 294.97869820000 -12.0000 147.0000 192.0000 0.0000 0.0000 0.0000 0.000000000

2.6.55 WAASCORRECTION SBAS

This command allows you to dictate how the card handles SBAS corrections. The receiver will automatically switch to Pseudorange Differential (RTCM or RTCA) or RTK if the appropriate corrections are received, regardless of the current setting.

To enable the position solution corrections, you must issue the WAASCORRECTION ENABLE command.

Abbreviated ASCII Syntax: **Message ID: 312**

WAASCORRECTION keyword [prn[mode]]

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively.	-	H	0
2	keyword	ENABLE	0	Card will use the SBAS corrections it receives.	Enum	4	H
		DISABLE	1	Card will not use the SBAS corrections it receives.			
3	prn	0		Card will use any PRN. (default)	ULong	4	H+4
		120-138		Card will use SBAS corrections only from this PRN.			
4	mode	NONE	0	Card will interpret Type 0 messages as they are intended (as do not use). (default)	Enum	4	H+8
		WAASTESTMODE	1	Card will interpret Type 0 messages as Type 2 messages.			
		EGNOSTESTMODE	2	Card will ignore the usual interpretation of Type 0 messages (as do not use) and continue.			

Abbreviated ASCII Example 1:
WAASCORRECTION ENABLE 0 WAASTESTMODE

Abbreviated ASCII Example 2:
WAASCORRECTION ENABLE 120 EGNOSTESTMODE

Chapter 3 Data Logs

3.1 Log Types

Refer to the LOG command, *see Page 71*, for details on requesting logs.

The receiver is capable of generating many different logs. These logs are divided into the following three types: Synchronous, asynchronous, and polled. The data for synchronous logs is generated on a regular schedule. Asynchronous data is generated at irregular intervals. If asynchronous logs were collected on a regular schedule, they would not output the most current data as soon as it was available. The data in polled logs is generated on demand. An example would be RXCONFIG. It would be polled because it changes only when commanded to do so. Therefore, it would not make sense to log this kind of data ONCHANGED, or ONNEW.

✉ The OEM4 family of receivers can handle 20 logs at a time. If you attempt to log more than 20 logs at a time, the receiver will respond with an Insufficient Resources error.

The following table outlines the log types and the valid triggers to use:

Table 33: Log Type Triggers

Type	Recommended Trigger	Illegal Trigger
Synch	ONTIME	ONNEW, ONCHANGED
Asynch	ONCHANGED	-
Polled	ONTIME	ONNEW, ONCHANGED

3.2 Logs By Function

Table 34 lists the logs by function while *Table 35 on Page 110* is an alphabetical listing of logs (repeated in *Table 36 on Page 114* with the logs in the order of their message IDs).

Table 34: Logs By Function Table

GENERAL RECEIVER CONTROL AND STATUS		
Logs	**Descriptions**	**Type**
COMCONFIG	Current COM port configuration	Polled
LOGLIST	List of system logs	Polled
PASSCOM1	Pass-through log, also PASSCOM2 and PASSCOM3	Asynch
PORTSTATS	COM port statistics	Polled
RXCONFIG	Receiver configuration status	Polled
RXHWLEVELS	Receiver hardware levels	Polled
RXSTATUS	Self-test status	Asynch
RXSTATUSEVENT	Status event indicator	Asynch
VALIDMODELS	Model and expiry date information for receiver	Asynch
VERSION	Receiver hardware and software version numbers	Polled

POSITION, PARAMETERS, AND SOLUTION FILTERING CONTROL		
Logs	**Descriptions**	**Type**
AVEPOS	Position averaging log	Asynch
BESTPOS [a]	Best position data	Synch
BESTXYZ	Cartesian coordinates position data	Synch
GPGGA	NMEA, fix and position data	Synch
GPGLL	NMEA, position data	Synch
GPGRS	NMEA, range residuals	Synch
GPGSA	NMEA, DOP information	Synch
GPGST	NMEA, measurement noise statistics	Synch
IONUTC	Ionospheric and UTC model information	Asynch
MATCHEDPOS [a]	Computed position	Asynch
MATCHEDXYZ	Cartesian coordinates computed position data	Asynch
MARKPOS	Position at time of markin event	Asynch
MARKTIME	Time of markin event	Asynch
OMNIHPPOS	OmniSTAR HP position data	Synch
PSRDOP	DOP of SVs currently tracking	Asynch
RTKPOS [a]	RTK low latency position	Synch
RTKXYZ	RTK cartesian coordinate position	Synch

a. The RTK system in the receiver provides two kinds of position solutions. The Matched RTK position is computed with buffered observations, so there is no error due to the extrapolation of base station measurements. This provides the highest accuracy solution possible at the expense of some latency which is affected primarily by the speed of the differential data link. The MATCHEDPOS log contains the matched RTK solution and can be generated for each processed set of base station observations. The RTKDATA log provides additional information about the matched RTK solution.

The Low-Latency RTK position is computed from the latest local observations and extrapolated base station observations. This supplies a valid RTK position with the lowest latency possible at the expense of some accuracy. The degradation in accuracy is reflected in the standard deviation and is summarized in the *Differential Positioning* section in the *GPS Overview* appendix in *Volume 1* of this manual. The amount of time that the base station observations are extrapolated is provided in the "differential age" field of the position log. The Low-Latency RTK system will extrapolate for 30 seconds. The RTKPOS log contains the Low-Latency RTK position when valid, and an "invalid" status when a low-latency RTK solution could not be computed. The BESTPOS log contains the low-latency RTK position when it is valid, and superior to the pseudorange-based position. Otherwise, it will contain the pseudorange-based position.

WAYPOINT NAVIGATION		
Logs	**Descriptions**	**Type**
BESTPOS	Best position data	Synch
BESTVEL[a]	Velocity data	Synch
GPRMB	NMEA, waypoint status	Synch
GPRMC	NMEA, navigation information	Synch
GPVTG	NMEA, track made good and speed	Synch
NAVIGATE	Navigation waypoint status	Synch
OMNIHPPOS	OmniSTAR HP position data	Synch
PSRPOS	Pseudorange position	Synch
PSRVEL[a]	Pseudorange velocity	Synch
PSRXYZ	Pseudorange cartesian coordinate position	Synch
RTKVEL[a]	RTK Velocity	Synch

CLOCK INFORMATION, STATUS, AND TIME		
Logs	**Descriptions**	**Type**
CLOCKMODEL	Range bias information	Synch
CLOCKSTEERING	Clock steering status	Asynch
GPZDA	NMEA, UTC time and date	Synch
TIME	Receiver time information	Synch
TIMESYNC	Synchronize time between receivers	Synch

a. The RTK velocity is computed from the latest local observations and extrapolated base station observations. This supplies a valid RTK velocity with the lowest latency possible at the expense of some accuracy. The degradation in accuracy is reflected in the standard deviation and is summarized in the *Differential Positioning* section in the *GPS Overview* appendix in *Volume 1* of this manual. The amount of time that the base station observations are extrapolated is provided in the "differential age" field of the velocity log. The Low-Latency RTK system will extrapolate for 30 seconds. The RTKVEL log contains the Low-Latency RTK velocity when valid, and an "invalid" status when a low-latency RTK solution could not be computed. The BESTVEL log contains the low-latency RTK velocity when it is valid, and superior to the pseudorange-based position. Otherwise, it will contain the pseudorange-based position.

   In the PSRVEL log the actual speed and direction of the GPSCard antenna over ground is provided. The GPSCard does not determine the direction a vessel, craft, or vehicle is pointed (heading), but rather the direction of motion of the GPS antenna relative to ground.

DIFFERENTIAL BASE STATION		
Logs	**Descriptions**	**Type**
ALMANAC	Current almanac information	Asynch
RANGE	Satellite range information	Synch
OMNIINFO	OmniSTAR configuration information	Synch
OMNISTAT	OmniSTAR status information	Synch
RAWOMNIFRAME	Raw OmniSTAR frame data	Asynch
RAWOMNIPACKET	Raw OmniSTAR data packet	Asynch

DIFFERENTIAL ROVER STATION		
Logs	**Descriptions**	**Type**
BESTPOS	Best position data	Synch
BESTVEL	Velocity data	Synch
GPGGA	NMEA, position fix data	Synch
MATCHEDPOS	Computed Position – Time Matched	Asynch
OMNIHPPOS	OmniSTAR HP position data	Synch
REFSTATION	Base station position and health	Asynch
RTKDATA	RTK related data such as baselines and satellite counts	Asynch
PSRPOS	Pseudorange position	Synch
PSRVEL	Pseudorange velocity	Synch
RTKPOS	RTK low latency position	Synch

POST PROCESSING DATA		
Logs	**Descriptions**	**Type**
GPSEPHEM	Decoded GPS ephemeris information	Asynch
IONUTC	Ionospheric and UTC model information	Asynch
RAWEPHEM	Raw ephemeris	Asynch
RANGE	Satellite range information	Synch
RTKDATA	RTK related data such as baselines and satellite counts.	Asynch
TIME	Receiver clock offset information	Synch

SATELLITE TRACKING AND CHANNEL CONTROL		
Logs	**Descriptions**	**Type**
ALMANAC	Current decoded almanac data	Asynch
GPALM	NMEA, almanac data	Synch
GPGSA	NMEA, SV DOP information	Synch
GPGSV	NMEA, satellite-in-view information	Synch
GPSEPHEM	Decoded GPS ephemeris information	Asynch
PSRDOP	DOP of SVs currently tracking	Asynch
RANGE	Satellite range information	Synch
RAWALM	Raw almanac	Asynch
RAWEPHEM	Raw ephemeris	Asynch
RAWGPSSUBFRAME	Raw subframe data	Asynch
RAWGPSWORD	Raw navigation word	Asynch
RAWWAASFRAME	Raw SBAS frame data	Asynch
SATVIS	Satellite visibility	Synch
SATXYZ	SV position in ECEF Cartesian coordinates	Synch
TRACKSTAT	Satellite tracking status	Synch
WAAS0	Remove PRN from the solution	Asynch
WAAS1	PRN mask assignments	Asynch
WAAS2	Fast correction slots 0-12	Asynch
WAAS3	Fast correction slots 13-25	Asynch
WAAS4	Fast correction slots 26-38	Asynch
WAAS5	Fast correction slots 39-50	Asynch
WAAS6	Integrity message	Asynch
WAAS7	Fast correction degradation	Asynch
WAAS9	GEO navigation message	Asynch
WAAS10	Degradation factor	Asynch
WAAS12	SBAS network time and UTC	Asynch
WAAS17	GEO almanac message	Asynch
WAAS18	IGP mask	Asynch
WAAS24	Mixed fast/slow corrections	Asynch
WAAS25	Long-term slow satellite corrections	Asynch
WAAS26	Ionospheric delay corrections	Asynch
WAAS27	SBAS service message	Asynch
WAASCORR	SBAS range corrections used	Synch

Table 35: OEM4 Family Logs in Alphabetical Order

NovAtel Format Logs		
Datatype	**Message ID**	**Description**
ALMANAC	73	Current almanac information
AVEPOS	172	Position averaging
BESTPOS	42	Best position data
BESTVEL	99	Velocity data
BESTXYZ	241	Cartesian coordinate position data
CLOCKMODEL	16	Current clock model matrices
CLOCKSTEERING	26	Clock steering status
CMRDATADESC	389	Base station description information
CMRDATAOBS	390	Base station satellite observation information
CMRDATAREF	391	Base station position information
COMCONFIG	317	Current COM port configuration
GPSEPHEM	7	GPS ephemeris data
IONUTC	8	Ionospheric and UTC model information
LOGLIST	5	A list of system logs
MARKPOS	181	Position at time of markin event
MARKTIME	231	Time of markin event
MATCHEDPOS	96	RTK Computed Position – Time Matched
MATCHEDXYZ	242	RTK Time Matched cartesian coordinate position data
NAVIGATE	161	Navigation waypoint status
OMNIHPPOS	495	OmniSTAR HP position data
OMNIINFO	509	OmniSTAR configuration information
OMNISTAT	510	OmniSTAR status information
PASSCOM1, PASSCOM2, PASSCOM3	233, 234, 235	Pass-through logs
PORTSTATS	72	COM port statistics
PSRDOP	174	DOP of SVs currently tracking
PSRPOS	47	Pseudorange position information
PSRVEL	100	Pseudorange velocity information
PSRXYZ	243	Pseudorange cartesian coordinate position information
RANGE	43	Satellite range information
RANGECMP	140	Compressed version of the RANGE log
RAWALM	74	Raw almanac
RAWEPHEM	41	Raw ephemeris

NovAtel Format Logs		
Datatype	**Message ID**	**Description**
RAWGPSSUBFRAME	25	Raw subframe data
RAWGPSWORD	407	Raw navigation word
RAWOMNIFRAME	441	Raw OmniSTAR frame data
RAWOMNIPACKET	457	Raw OmniSTAR data packet
RAWWAASFRAME	287	Raw SBAS frame data
REFSTATION	175	Base station position and health
RTCADATA1	392	Type 1 Differential GPS Corrections
RTCADATAEPHEM	393	Type 7 Ephemeris and Time Information
RTCADATAOBS	394	Type 7 Base Station Observations
RTCADATAREF	395	Type 7 Base Station Parameters
RTCMDATA1	396	Type 1 Differential GPS Corrections
RTCMDATA3	402	Type 3 Base Station Parameters
RTCMDATA9	404	Type 9 Partial Differential GPS Corrections
RTCMDATA15	397	Type 15 Ionospheric Corrections
RTCMDATA16	398	Type 16 Special Message
RTCMDATA1819	399	Type18 and Type 19 Raw Measurements
RTCMDATA2021	400	Type 20 and Type 21 Measurement Corrections
RTCMDATA22	401	Type 22 Extended Base Station Parameters
RTCMDATA59	403	Type 59N-0 NovAtel Proprietary: RT20 Differential
RTKDATA	215	RTK specific information
RTKPOS	141	RTK low latency position data
RTKVEL	216	RTK velocity
RTKXYZ	244	RTK cartesian coordinate position data
RXCONFIG	128	Receiver configuration status
RXHWLEVELS	195	Receiver hardware levels
RXSTATUS	93	Self-test status
RXSTATUSEVENT	94	Status event indicator
SATVIS	48	Satellite visibility
SATXYZ	270	SV position in ECEF Cartesian coordinates
TIME	101	Receiver time information
TIMESYNC	492	Synchronize time between receivers
TRACKSTAT	83	Satellite tracking status
VALIDMODELS	206	Model and expiry date information for receiver
VERSION	37	Receiver hardware and software version numbers

NovAtel Format Logs		
Datatype	**Message ID**	**Description**
WAAS0	290	Remove PRN from the solution
WAAS1	291	PRN mask assignments
WAAS2	296	Fast correction slots 0-12
WAAS3	301	Fast correction slots 13-25
WAAS4	302	Fast correction slots 26-38
WAAS5	303	Fast correction slots 39-50
WAAS6	304	Integrity message
WAAS7	305	Fast correction degradation
WAAS9	306	GEO navigation message
WAAS10	292	Degradation factor
WAAS12	293	SBAS network time and UTC
WAAS17	294	GEO almanac message
WAAS18	295	IGP mask
WAAS24	297	Mixed fast/slow corrections
WAAS25	298	Long term slow satellite corrections
WAAS26	299	Ionospheric delay corrections
WAAS27	300	SBAS service message
WAASCORR	313	SBAS range corrections used
CMR Format Logs [a]		
CMRDESC	310	Base station description information
CMROBS	103	Base station satellite observation information
CMRREF	105	Base station position information
RTCA Format Logs [a]		
RTCA1	10	Type 1 Differential GPS Corrections
RTCAEPHEM	347	Type 7 Ephemeris and Time Information
RTCAOBS	6	Type 7 Base Station Observations
RTCAREF	11	Type 7 Base Station Parameters
RTCM Format Logs [a]		
RTCM1	107	Type 1 Differential GPS Corrections
RTCM3	117	Type 3 Base Station Parameters
RTCM9	275	Type 9 Partial Differential GPS Corrections
RTCM15	307	Type 15 Ionospheric Corrections
RTCM16	129	Type16 Special Message

Datatype	Message ID	Description
RTCM Format Logs [a]		
RTCM16T	131	Type16T Special Text Message
RTCM1819	260	Type18 and Type 19 Raw Measurements
RTCM2021	374	Type 20 and Type 21 Measurement Corrections
RTCM22	118	Type 22 Extended Base Station Parameters
RTCM59	116	Type 59N-0 NovAtel Proprietary: RT20 Differential
NMEA Format Logs		
GPALM	217	Almanac Data
GPGGA	218	GPS Fix Data and Undulation
GPGGARTK	259	GPS Fix Data
GPGLL	219	Geographic Position - latitude/longitude
GPGRS	220	GPS Range Residuals for Each Satellite
GPGSA	221	GPS DOP and Active Satellites
GPGST	222	Pseudorange Measurement Noise Statistics
GPGSV	223	GPS Satellites in View
GPRMB	224	Generic Navigation Information
GPRMC	225	GPS Specific Information
GPVTG	226	Track Made Good and Ground Speed
GPZDA	227	UTC Time and Date

a. CMR, RTCA, and RTCM logs may be logged with an A or B extension to give an ASCII or Binary output with a NovAtel header followed by Hex or Binary data respectively

Table 36: OEM4 Family Logs in Order of their Message IDs

NovAtel Format Logs		
Message ID	**Datatype**	**Description**
5	LOGLIST	A list of system logs
7	GPSEPHEM	GPS ephemeris data
8	IONUTC	Ionospheric and UTC model information
16	CLOCKMODEL	Current clock model matrices
25	RAWGPSSUBFRAME	Raw subframe data
26	CLOCKSTEERING	Clock steering status
37	VERSION	Receiver hardware and software version numbers
41	RAWEPHEM	Raw ephemeris
42	BESTPOS	Best position data
43	RANGE	Satellite range information
47	PSRPOS	Pseudorange position information
48	SATVIS	Satellite visibility
72	PORTSTATS	COM port statistics
73	ALMANAC	Current almanac information
74	RAWALM	Raw almanac
83	TRACKSTAT	Satellite tracking status
93	RXSTATUS	Self-test status
94	RXSTATUSEVENT	Status event indicator
96	MATCHEDPOS	RTK Computed Position – Time Matched
99	BESTVEL	Velocity data
100	PSRVEL	Pseudorange velocity information
101	TIME	Receiver time information
128	RXCONFIG	Receiver configuration status
140	RANGECMP	Compressed version of the RANGE log
141	RTKPOS	RTK low latency position data
161	NAVIGATE	Navigation waypoint status
172	AVEPOS	Position averaging
174	PSRDOP	DOP of SVs currently tracking
175	REFSTATION	Base station position and health
181	MARKPOS	Position at time of markin event
195	RXHWLEVELS	Receiver hardware levels
206	VALIDMODELS	Model and expiry date information for receiver
215	RTKDATA	RTK specific information

NovAtel Format Logs		
Message ID	**Datatype**	**Description**
216	RTKVEL	RTK velocity
231	MARKTIME	Time of markin event
233	PASSCOM1, PASSCOM2, PASSCOM3	Pass-through logs
241	BESTXYZ	Cartesian coordinate position data
242	MATCHEDXYZ	RTK Time Matched cartesian coordinate position data
243	PSRXYZ	Pseudorange cartesian coordinate position information
244	RTKXYZ	RTK cartesian coordinate position data
270	SATXYZ	SV position in ECEF Cartesian coordinates
287	RAWWAASFRAME	Raw SBAS frame data
290	WAAS0	Remove PRN from the solution
291	WAAS1	PRN mask assignments
292	WAAS10	Degradation factor
293	WAAS12	SBAS network time and UTC
294	WAAS17	GEO almanac message
295	WAAS18	IGP mask
296	WAAS2	Fast correction slots 0-12
297	WAAS24	Mixed fast/slow corrections
298	WAAS25	Long term slow satellite corrections
299	WAAS26	Ionospheric delay corrections
300	WAAS27	SBAS service message
301	WAAS3	Fast correction slots 13-25
302	WAAS4	Fast correction slots 26-38
303	WAAS5	Fast correction slots 39-50
304	WAAS6	Integrity message
305	WAAS7	Fast correction degradation
306	WAAS9	GEO navigation message
313	WAASCORR	SBAS range corrections used
317	COMCONFIG	Current COM port configuration
389	CMRDATADESC	Base station description information
390	CMRDATAOBS	Base station satellite observation information
391	CMRDATAREF	Base station position information
392	RTCADATA1	Type 1 Differential GPS Corrections
393	RTCADATAEPHEM	Type 7 Ephemeris and Time Information

NovAtel Format Logs		
Message ID	**Datatype**	**Description**
394	RTCADATAOBS	Type 7 Base Station Observations
395	RTCADATAREF	Type 7 Base Station Parameters
396	RTCMDATA1	Type 1 Differential GPS Corrections
397	RTCMDATA15	Type 15 Ionospheric Corrections
398	RTCMDATA16	Type 16 Special Message
399	RTCMDATA1819	Type18 and Type 19 Raw Measurements
400	RTCMDATA2021	Type 20 and Type 21 Measurement Corrections
401	RTCMDATA22	Type 22 Extended Base Station Parameters
402	RTCMDATA3	Type 3 Base Station Parameters
403	RTCMDATA59	Type 59N-0 NovAtel Proprietary: RT20 Differential
404	RTCMDATA9	Type 9 Partial Differential GPS Corrections
407	RAWGPSWORD	Raw navigation word
441	RAWOMNIFRAME	Raw OmniSTAR frame data
457	RAWOMNIPACKET	Raw OmniSTAR data packet
492	TIMESYNC	Synchronize time between receivers
495	OMNIHPPOS	OmniSTAR HP position data
509	OMNIINFO	OmniSTAR configuration information
510	OMNISTAT	OmniSTAR status information
CMR Format Logs [a]		
103	CMROBS	Base station satellite observation information
105	CMRREF	Base station position information
310	CMRDESC	Base station description information
RTCA Format Logs [a]		
6	RTCAOBS	Type 7 Base Station Observations
10	RTCA1	Type 1 Differential GPS Corrections
11	RTCAREF	Type 7 Base Station Parameters
347	RTCAEPHEM	Type 7 Ephemeris and Time Information
RTCM Format Logs [a]		
107	RTCM1	Type 1 Differential GPS Corrections
116	RTCM59	Type 59N-0 NovAtel Proprietary: RT20 Differential
117	RTCM3	Type 3 Base Station Parameters
118	RTCM22	Type 22 Extended Base Station Parameters
129	RTCM16	Type16 Special Message

Message ID	Datatype	Description
RTCM Format Logs [a]		
131	RTCM16T	Type16T Special Text Message
260	RTCM1819	Type18 and Type 19 Raw Measurements
275	RTCM9	Type 9 Partial Differential GPS Corrections
307	RTCM15	Type 15 Ionospheric Corrections
374	RTCM2021	Type 20 and Type 21 Measurement Corrections
NMEA Format Data Logs		
217	GPALM	Almanac Data
218	GPGGA	GPS Fix Data and Undulation
219	GPGLL	Geographic Position - latitude/longitude
220	GPGRS	GPS Range Residuals for Each Satellite
221	GPGSA	GPS DOP and Active Satellites
222	GPGST	Pseudorange Measurement Noise Statistics
223	GPGSV	GPS Satellites in View
224	GPRMB	Generic Navigation Information
225	GPRMC	GPS Specific Information
226	GPVTG	Track Made Good and Ground Speed
227	GPZDA	UTC Time and Date
259	GPGGARTK	GPS Fix Data

a. CMR, RTCA, and RTCM logs may be logged with an A or B extension to give an ASCII or Binary output with a NovAtel header followed by Hex or Binary data respectively

3.3 MiLLennium GPSCard Compatibility

Table 37, MiLLennium OEM3 Log Comparison on Page 118 shows the MiLLennium logs that are comparable to current OEM4 family logs.

Table 37: MiLLennium OEM3 Log Comparison

MiLLennium Log	Comparable OEM4/EURO4 Log
ALM	ALMANAC and IONUTC
BSL	RTKDATA
CDS	PORTSTATS
CLK	CLOCKMODEL
CLM	CLOCKMODEL
CMR	CMR
COM1	PASSCOM1
COM2	PASSCOM2
DOP	PSRDOP
ETS	TRACKSTAT
FRM	RAWGPSSUBFRAME and RAWWAASFRAME
FRW	RAWGPSWORD
GGB	Not currently supported.
GP∗ (NMEA logs)	Same as MiLLennium.
MKP	MARKPOS
MKT	MARKTIME
NAV	NAVIGATE
PAV	AVEPOS
POS	BESTPOS
PRTK	BESTPOS and RTKPOS
PVA	BESTXYZ, MATCHEDXYZ, PSRXYZ and RTKXYZ
PXY	BESTXYZ, MATCHEDXYZ, PSRXYZ and RTKXYZ
RAL	RAWALM
RAS	RAWALM
RBT	Not currently supported.
RCCA	RXCONFIG, COMCONFIG and LOGLIST
RCS	RXSTATUS and VERSION
REP	RAWEPHEM
RGE	RANGE and RANGECMP
RPS	REFSTATION
RTCA	RTCA
RTK	MATCHEDPOS
RTKO	RTKDATA
RTCM	RTCM
RVS	RXSTATUS
SAT	SATVIS
SBT	Not currently supported.
SPH	PSRVEL
SVD	SATXYZ
TM1	TIME
VER	VERSION
VLH	PSRVEL
WAL	Not currently supported.
WRC	Not currently supported.

3.4 Log Reference

3.4.1 ALMANAC Decoded Almanac

This log contains the decoded almanac parameters from subframes four and five as received from the satellite with the parity information removed and appropriate scaling applied. Multiple messages are transmitted, one for each SV almanac collected. For more information on Almanac data, refer to the GPS SPS Signal Specification. (See the appendix on *Standards and References* in *Volume 1* of this manual.)

The OEM4 family of receives will automatically save almanacs in their non-volatile memory (NVM), therefore creating an almanac boot file is not necessary.

Structure: **Message ID: 73**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#messages	The number of satellite PRN almanac messages to follow.	Long	4	H
3	PRN	Satellite PRN number for current message, dimensionless	Ulong	4	H+4
4	week	Almanac reference week (GPS week number)	Ulong	4	H+8
5	seconds	Almanac reference time, seconds into the week	Double	8	H+12
6	ecc	Eccentricity, dimensionless	Double	8	H+20
7	$\dot{\omega}$	Rate of right ascension, radians/second	Double	8	H+28
8	ω_0	Right ascension, radians	Double	8	H+36
9	ω	Argument of perigee, radians	Double	8	H+44
10	M_0	Mean anomaly of reference time, radians	Double	8	H+52
11	a_{fo}	Clock aging parameter, seconds	Double	8	H+60
12	a_{f1}	Clock aging parameter, seconds/second	Double	8	H+68
13	N	Corrected mean motion, radians/second	Double	8	H+76
14	A	Semi-major axis, meters	Double	8	H+84
15	incl-angle	Angle of inclination relative to 0.3 π, radians	Double	8	H+92
16	SV config	Satellite configuration	Ulong	4	H+100
17	health-prn	SV health from subframe 4 or 5	Ulong	4	H+104
18	health-alm	SV health from almanac	Ulong	4	H+108
19	antispoof	Anti-spoofing on: 0 = FALSE 1 = TRUE	Enum	4	H+112
20...	Next PRN offset = H + 4 + (#messages x 112)				
21	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H + 4 + (112 x #messages)
22	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log almanaca onchanged

ASCII Example:

```
#ALMANACA,COM1,0,64.5,SATTIME,1048,508014.000,00000028,42f1,0;
31,
1,1049,61440.0,5.10788e-03,-7.8974718e-09,-4.5666114e-01,-1.7361153e+00,
-9.1286238e-01,1.1730194e-04,0.00000000,1.4584974e-04,2.6560862e+07,
1.7532921e-02,1,0,0,FALSE,
2,1049,61440.0,1.95422e-02,-8.0917656e-09,1.5495060e+00,-2.1449823e+00,
2.5208892e+00,-1.7547607e-04,-3.6379788e-12,1.4585339e-04,2.6560419e+07,
-8.2930836e-03,1,0,0,FALSE
…
30,1049,61440.0,5.61333e-03,-7.9889042e-09,1.5991652e+00,1.4759191e+00,
2.7901058e+00,-9.5367432e-06,0.00000000,1.4585534e-04,2.6560183e+07,
1.9534287e-03,1,0,0,FALSE,
…
31,1049,61440.0,9.47952e-03,-8.0917656e-09,2.6421445e+00,8.0564663e-01,
1.9109259e-01,3.7193298e-05,0.00000000,1.4585948e-04,2.6559680e+07,
8.8922949e-03,1,0,0,TRUE*fc97918d
```

3.4.2 AVEPOS Position Averaging

When position averaging is underway, the various fields in the AVEPOS log contain the parameters being used in the position averaging process.

See the description of the POSAVE command on *Page 78*. See also the sections on *Height Relationships* and *Pseudorange Algorithms* in the *GPS Overview* appendix in *Volume 1* of this manual.

✉ All quantities are referenced to the geoid (average height above sea level), regardless of the use of the DATUM or USERDATUM commands, except for the height parameter (field 6). The relation between the geoid and the WGS84 ellipsoid is the geoidal undulation, and can be obtained from the PSRPOS log.

Structure: **Message ID: 172**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	lat	Average WGS84 latitude (degrees)	Double	8	H
3	lon	Average WGS84 longitude (degrees)	Double	8	H+8
4	ht	Average height above sea level, or geoid (m)	Double	8	H+16
5	lat σ	Estimated average standard deviation of latitude solution element, in meters	Float	4	H+24
6	lon σ	Estimated average standard deviation of longitude solution element, in meters	Float	4	H+28
7	hgt σ	Estimated average standard deviation of height solution element, in meters	Float	4	H+32
8	posave	Position averaging status (see *Table 38*)	Enum	4	H+36
9	ave time	Elapsed time of averaging (s)	Ulong	4	H+40
10	samples	Number of samples in the average	Ulong	4	H+44
11	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+48
12	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log aveposa onchanged

ASCII Example:

```
#AVEPOSA,COM1,0,65.0,FINESTEERING,1058,272851.000,80000000,3a72,32800;
51.11635672780,-114.03824509533,1046.6118,5.0299,3.4216,8.0996,INPROGRESS,
13800,24*a2f7b21b
```

Table 38: Position Averaging Status

Binary	ASCII	Description
0	OFF	Card is not averaging
1	INPROGRESS	Averaging is in progress
2	COMPLETE	Averaging is complete

3.4.3 BESTPOS Best Position

This log contains the best available combined GPS and inertial navigation system (INS - if available) position computed by the receiver. In addition, it reports several status indicators, including differential age, which is useful in predicting anomalous behavior brought about by outages in differential corrections. A differential age of 0 indicates that no differential correction was used.

With the system operating in an RTK mode, this log will reflect the latest low-latency solution for up to 30 seconds after reception of the last base station observations. After this 30 second period, the position reverts to the best solution available; the degradation in accuracy is reflected in the standard deviation fields. If the system is not operating in an RTK mode, pseudorange differential solutions continue for the time specified in the DGPSTIMEOUT command, see *Page 57*.

See also the table footnote for position logs on *Page 106* as well as the MATCHEDPOS, PSRPOS and RTKPOS logs, on *Pages 159, 173 and 214* respectively.

Structure: **Message ID: 42**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	lat	Latitude	Double	8	H+8
5	lon	Longitude	Double	8	H+16
6	hgt	Height above mean sea level	Double	8	H+24
7	undulation	Undulation	Float	4	H+32
8	datum id#	Datum ID number (see *Chapter 2, Table 17, Datum Transformation Parameters on Page 54*)	Enum	4	H+36
9	lat σ	Latitude standard deviation	Float	4	H+40
10	lon σ	Longitude standard deviation	Float	4	H+44
11	hgt σ	Height standard deviation	Float	4	H+48
12	stn id	Base station ID	Char[4]	4	H+52
13	diff_age	Differential age	Float	4	H+56
14	sol_age	Solution age in seconds	Float	4	H+60
15	#obs	Number of observations tracked	Uchar	1	H+64
16	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+65
17	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+66
18	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+67
19	Reserved		Uchar	1	H+68
20			Uchar	1	H+69
21			Uchar	1	H+70
22			Uchar	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log bestposa ontime 1

See *Section 2.1, Command Formats on Page 22* for more examples of log requests.

ASCII Example:

```
#BESTPOSA,COM1,0,62.5,FINESTEERING,1036,484878.000,00000028,63e2,0;
SOL_COMPUTED,SINGLE,51.11629893124,-114.03820302746,1052.3434,
-16.271287293,61,19.6934,13.1515,23.8561,"",0.0,60.000,10,10,0,0,
0,0,0,0*1051ada9
```

Table 39: Position or Velocity Type

Position Type (binary)	Position Type (ASCII)	Description
0	NONE	No solution
1	FIXEDPOS	Position has been fixed by the FIX POSITION command
2	FIXEDHEIGHT	Position has been fixed by the FIX HEIGHT, or FIX AUTO, command
3	FIXEDVEL	Position has been fixed by the FIX VELOCITY command
8	DOPPLER_VELOCITY	Velocity has been computed using instantaneous Doppler measurements
16	SINGLE	Single point position
17	PSRDIFF	Pseudorange differential solution
18	WAAS	Solution calculated using corrections from a Satellite Based Augmentation System (SBAS)
20	OMNISTAR	OmniSTAR VBS position (L1 sub-meter) [a]
32	L1_FLOAT	Floating L1 ambiguity solution
33	IONOFREE_FLOAT	Floating ionospheric-free ambiguity solution
34	NARROW_FLOAT	Floating narrow-lane ambiguity solution
48	L1_INT	Integer L1 ambiguity solution
49	WIDE_INT	Integer wide-lane ambiguity solution
50	NARROW_INT	Integer narrow-lane ambiguity solution
51	RTK_DIRECT_INS	RTK status where the RTK filter is directly initialized from the INS filter [b]
52	INS	INS calculated position corrected for the antenna [b]
64	OMNISTAR_HP	OmniSTAR HP position (L1/L2 decimeter) [a]

a. In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details.
b. Output only by the BESTPOS and BESTVEL logs when using an inertial navigation system such as NovAtel's Black Diamond System (BDS). Please visit our website, refer to your BDS Users' Guide, or contact NovAtel for more information.

Table 40: Solution Status

Solution Status (Binary)	Solution Status (ASCII)	Description
0	SOL_COMPUTED	Solution computed
1	INSUFFICIENT_OBS	Insufficient observations
2	NO_CONVERGENCE	No convergence
3	SINGULARITY	Singularity at parameters matrix
4	COV_TRACE	Covariance trace exceeds maximum (trace > 1000 m)
5	TEST_DIST	Test distance exceeded (maximum of 3 rejections if distance > 10 km)
6	COLD_START	Not yet converged from cold start
7	V_H_LIMIT	Height or velocity limits exceeded (in accordance with COCOM export licensing restrictions)
8	VARIANCE	Variance exceeds limits
9	RESIDUALS	Residuals are too large
10	DELTA_POS	Delta position is too large
11	NEGATIVE_VAR	Negative variance
12	Reserved	
13	INTEGRITY_WARNING	Large residuals make position unreliable

3.4.4 BESTVEL Best Available Velocity Data

This log contains the best available velocity information computed by the receiver. In addition, it reports a velocity status indicator, which is useful in indicating whether or not the corresponding data is valid. The velocity measurements sometimes have a latency associated with them. The time of validity is the time tag in the log minus the latency value. See also the table footnote for velocity logs on *Page 107*.

The velocity is typically computed from the average change in pseudorange over the time interval or the RTK Low Latency filter. As such, it is an average velocity based on the time difference between successive position computations and not an instantaneous velocity at the BESTVEL time tag. The velocity latency to be subtracted from the time tag will normally be 1/2 the time between filter updates. Under default operation, the positioning filters are updated at a rate of 2 Hz. This translates into a velocity latency of 0.25 second. The latency can be reduced by increasing the update rate of the positioning filter being used by requesting the BESTVEL or BESTPOS messages at a rate higher than 2 Hz. For example, a logging rate of 10 Hz would reduce the velocity latency to 0.005 seconds. For integration purposes, the velocity latency should be applied to the record time tag.

A valid solution with a latency of 0.0 indicates that the instantaneous Doppler measurement was used to calculate velocity.

Structure: **Message ID: 99**

Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	latency	A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results.	Float	4	H+8
5	age	Differential age	Float	4	H+12
6	hor spd	Horizontal speed over ground, in meters per second	Double	8	H+16
7	trk gnd	Actual direction of motion over ground (track over ground) with respect to True North, in degrees	Double	8	H+24
8	vert spd	Vertical speed, in meters per second, where positive values indicate increasing altitude (up) and negative values indicate decreasing altitude (down)	Double	8	H+32
9	Reserved		Float	4	H+40
10	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
11	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log bestvela ontime 1

ASCII Example:

```
#BESTVELA,COM1,0,62.5,FINESTEERING,1049,247755.000,00000128,f7e3,0;
SOL_COMPUTED,DOPPLER_VELOCITY,0.250,0.000,0.1744,333.002126,0.3070,
6.0082*dfdc635c
```

3.4.5 BESTXYZ Best Available Cartesian Position and Velocity

This log contains the receiver's best available position and velocity in ECEF coordinates. The position and velocity status fields indicate whether or not the corresponding data is valid. See *Figure 6, Page 127* for a definition of the ECEF coordinates.

See also the BESTPOS and BESTVEL logs, on *Pages 122 and 125* respectively.

✉ These quantities are always referenced to the WGS84 ellipsoid, regardless of the use of the DATUM or USERDATUM commands.

Structure: **Message ID: 241**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	P-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	P-X	Position X-coordinate (m)	Double	8	H+8
5	P-Y	Position Y-coordinate (m)	Double	8	H+16
6	P-Z	Position Z-coordinate (m)	Double	8	H+24
7	P-X σ	Standard deviation of P-X (m)	Float	4	H+32
8	P-Y σ	Standard deviation of P-Y (m)	Float	4	H+36
9	P-Z σ	Standard deviation of P-Z (m)	Float	4	H+40
10	V-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H+44
11	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+48
12	V-X	Velocity vector along X-axis (m/s)	Double	8	H+52
13	V-Y	Velocity vector along Y-axis (m/s)	Double	8	H+60
14	V-Z	Velocity vector along Z-axis (m/s)	Double	8	H+68
15	V-X σ	Standard deviation of V-X (m/s)	Float	4	H+76
16	V-Y σ	Standard deviation of V-Y (m/s)	Float	4	H+80
17	V-Z σ	Standard deviation of V-Z (m/s)	Float	4	H+84
18	stn ID	Base station identification	Char[4]	4	H+88
19	V-latency	A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results.	Float	4	H+92
20	diff_age	Differential age	Float	4	H+96
21	sol_age	Solution age in seconds	Float	4	H+100
22	#obs	Number of observations tracked	Uchar	1	H+104
23	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+105
24	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+106
25	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+107

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
26	Reserved		Char	1	H+108
27			Char	1	H+109
28			Char	1	H+110
29			Char	1	H+111
30	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+112
31	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log bestxyza ontime 1

ASCII Example:

```
#BESTXYZA,COM1,0,72.5,FINESTEERING,1058,404080.000,00000028,06e1,37;
SOL_COMPUTED,SINGLE,-1634502.769,3664601.068,4942471.986,26.3812,
34.7726,38.3204,SOL_COMPUTED,DOPPLER_VELOCITY,0.195,0.496,-0.254,3.1532,
4.1561,4.5802,"",0.250,0.000,0.000,6,6,0,0,0,0,0,0*f370ee18
```

Figure 6: The WGS84 ECEF Coordinate System

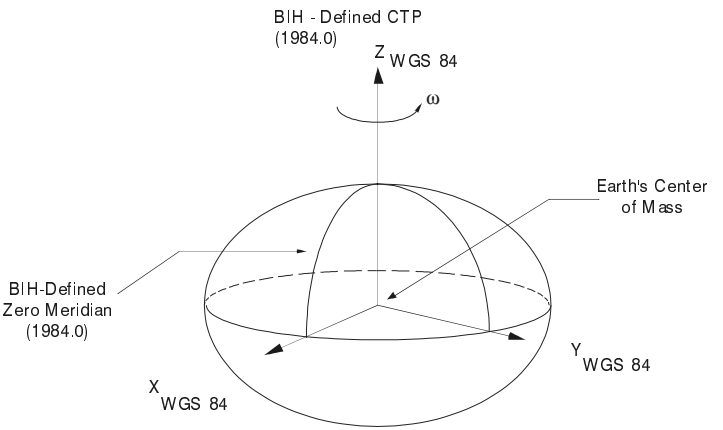
- Definitions - *

Origin = Earth's center of mass

Z-Axis = Parallel to the direction of the Conventional Terrestrial Pole (CTP) for polar motion, as defined by the Bureau International de l'Heure (BIH) on the basis of the coordinates adopted for the BIH stations.

X-Axis = Intersection of the WGS 84 Reference Meridian Plane and the plane of the CTP's Equator, the Reference Meridian being parallel to the Zero Meridian defined by the BIH on the basis of the coordinates adopted for the BIH stations.

Y-Axis = Completes a right-handed, earth-centered, earth-fixed (ECEF) orthogonal coordinate system, measured in the plane of the CTP Equator, 90i East of the X-Axis.



* Analogous to the BIH Defined Conventional Terrestrial System (CTS), or BTS, 1984.0.

3.4.6 CLOCKMODEL Current Clock Model Status

The CLOCKMODEL log contains the current clock-model status of the receiver.

Monitoring the CLOCKMODEL log will allow you to determine the error in your GPSCard receiver reference oscillator as compared to the GPS satellite reference.

All logs report GPS time not corrected for local receiver clock error. To derive the closest GPS time, subtract the clock offset shown in the CLOCKMODEL log (field #4) from the GPS time reported.

The following symbols are used throughout this section:

B = range bias (m)
BR = range bias rate (m/s)
SAB = Gauss-Markov process representing range bias error due to SA clock dither (m)

The standard clock model now used is as follows:

clock parameters array = [B BR SAB]

covariance matrix =

$$\begin{bmatrix} \sigma_B^2 & \sigma_B \sigma_{BR} & \sigma_B \sigma_{SAB} \\ \sigma_{BR} \sigma_B & \sigma_{BR}^2 & \sigma_{BR} \sigma_{SAB} \\ \sigma_{SAB} \sigma_B & \sigma_{SAB} \sigma_{BR} & \sigma_{SAB}^2 \end{bmatrix}$$

Table 41: Clock Model Status

Clock Status (Binary)	Clock Status (ASCII)	Description
0	VALID	The clock model is valid
1	CONVERGING	The clock model is near validity
2	ITERATING	The clock model is iterating towards validity
3	INVALID	The clock model is not valid
4	ERROR	Clock model error

Structure: **Message ID: 16**

Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	clock status	Clock model status as computed from current measurement data, see *Table 41, Clock Model Status on Page 128*	Enum	4	H
3	reject	Number of rejected range bias measurements	Ulong	4	H+4
4	noise time	GPS time of last noise addition	GPSec	4	H+8
5	update time	GPS time of last update	GPSec	4	H+12
6	parameters	Clock correction parameters (a 1x3 array of length 3), listed left-to-right	Double	8	H+16
7				8	H+24
8				8	H+32
9	cov data	Covariance of the straight line fit (a 3x3 array of length 9), listed left-to-right by rows	Double	8	H+40
10				8	H+48
11				8	H+56
12				8	H+64
13				8	H+72
14				8	H+80
15				8	H+88
16				8	H+96
17				8	H+104
18	range bias	Last instantaneous measurement of the range bias (meters)	Double	8	H+112
19	range bias rate	Last instantaneous measurement of the range bias rate (m/s)	Double	8	H+120
20	change	Is there a change in the constellation? 0 = FALSE 1 = TRUE	Enum	4	H+128
21	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+132
22	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log clockmodela ontime 1

ASCII Example:

```
#CLOCKMODELA,COM1,0,55.0,FINESTEERING,1048,497224.500,00000128,879e,0;
VALID,0,497224.500,497224.500,6.59596736e-01,-9.29232987e-02,
-3.40165918e+00,4.63006778e+02,1.19966616e+00,-4.50054493e+02,
1.19966616e+00,2.15673338e-01,-7.90256149e-01,-4.50054493e+02,
-7.90256149e-01,4.76057122e+02,-2.902,-1.03659974e-01,FALSE*f33b4465
```

3.4.7 CLOCKSTEERING Clock Steering Status

The CLOCKSTEERING log is used to monitor the current state of the clock steering process. All oscillators have some inherent drift. By default the receiver attempts to steer the receiver's clock to accurately match GPS time. If for some reason this is not desired, this behavior can be disabled using the CLOCKADJUST command, see *Page 46*.

✉ If the CLOCKADJUST command is ENABLED, and the receiver is configured to use an external reference frequency (set in the EXTERNALCLOCK command, see *Page 62*, for an external clock - TCXO, OCXO, RUBIDIUM, CESIUM, or USER), then the clock steering process will take over the VARF output pins and may conflict with a previously entered FREQUENCYOUT command, see *Page 67*.

Structure: **Message ID: 26**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	source	Clock source, see *Table 42, Clock Source on Page 131*.	Enum	4	H
3	steeringstate	Steering state, see *Table 43, Steering State on Page 131*.	Enum	4	H+4
4	period	Period of the FREQUENCYOUT signal used to control the oscillator, refer to the FREQUENCYOUT command. This value is set using the CLOCKCALIBRATE command.	Ulong	4	H+8
5	pulsewidth	Current pulse width of the FREQUENCYOUT signal. The starting point for this value is set using the CLOCKCALIBRATE command. The clock steering loop will continuously adjust this value in an attempt to drive the receiver clock offset and drift terms to zero.	Ulong	4	H+12
6	bandwidth	The current band width of the clock steering tracking loop in Hz. This value is set using the CLOCKCALIBRATE command.	Float	4	H+16
7	slope	The current clock drift change in m/s/bit for a 1 LSB pulse width. This value is set using the CLOCKCALIBRATE command.	Float	4	H+20
8	offset	The last valid receiver clock offset computed (m). It is the same as Field # 18 of the CLOCKMODEL log, see *Page 128*.	Double	8	H+24
9	driftrate	The last valid receiver clock drift rate received (m/s). It is the same as Field # 19 of the CLOCKMODEL log.	Double	8	H+32
10	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+40
11	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log clocksteering onchanged

ASCII Example:

```
#CLOCKSTEERINGA,COM1,0,65.0,FINESTEERING,1058,272851.000,80000000,3a72,32800;
INTERNAL,FIRST_ORDER,4400,2135.294,0.03,-2.0,-2.092,-1.03659974e-01*a2f7b21b
```

Table 42: Clock Source

Binary	ASCII	Description
0	INTERNAL	The receiver is currently steering its internal VCTCXO using an internal VARF signal.
1	EXTERNAL	The receiver is currently steering an external oscillator using the external VARF signal.

Table 43: Steering State

Binary	ASCII	Description
0	FIRST_ORDER	Upon startup, the clock steering task will adjust the VARF pulse width to reduce the receiver clock drift rate to below 1 ms using a 1st order control loop. This is the normal startup state of the clock steering loop.
1	SECOND_ORDER	Once the receiver has reduced the clock drift to below 1 m/s, it enters a second order control loop and will attempt to reduce the receiver clock offset to zero. This is the normal runtime state of the clock steering process.
2	CALIBRATE_HIGH [a]	This state corresponds to when the calibration process is measuring at the "High" pulse width setting.
3	CALIBRATE_LOW [a]	This state corresponds to when the calibration process is measuring at the "Low" pulse width setting.
4	CALIBRATE_CENTER [b]	This state corresponds to the "Center" calibration process. Once the center has been found, the modulus pulse width, center pulse width, loop bandwidth, and measured slope values are saved in NVM and are used from now on for the currently selected oscillator (INTERNAL or EXTERNAL).

a. These states will only be seen if you force the receiver to do a clock steering calibration using the CLOCKCALIBRATE command, see *Page 47*. With the CLOCKCALIBRATE command, you can force the receiver to calibrate the slope and center pulse width, of the currently selected oscillator, to steer. The receiver will measure the drift rate at several "High" and "Low" pulse width settings.

b. After the receiver has measured the "High" and "Low" pulse width setting, the calibration process enters a "Center calibration" process where it attempts to find the pulse width required to zero the clock drift rate.

3.4.8 CMR Standard Logs *RTK*

CMROBS BASE STATION SATELLITE OBSERVATION INFORMATION

Message ID: 103

CMRREF BASE STATION POSITION INFORMATION

Message ID: 105

CMRDESC BASE STATION DESCRIPTION INFORMATION

Message ID: 310

The Compact Measurement Record (CMR) Format, is a standard communications protocol used in Real-Time Kinematic (RTK) systems to transfer GPS carrier phase and code observations from a base station to one or more rover stations.

✉ The above messages can be logged with an A or B suffix for an ASCII or Binary output with a NovAtel header followed by Hex or Binary raw data respectively.

See the chapter on *Message Formats* in *Volume 1* of this manual for more information on CMR standard logs.

Example Input:

```
interfacemode com2 none CMR
fix position 51.113 -114.044 1059.4
log com2 cmrobs ontime 2
log com2 cmrref ontime 10
log com2 cmrdesc ontime 10 5
```

3.4.9 CMRDATADESC Base Station Description RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on CMR standard logs.

Structure: **Message ID: 389**

Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	CMR header	Synch character for the message	Ulong	4	H
3		Message status	Ulong	4	H+4
4		CMR message type	Ulong	4	H+8
5		Message body length	Ulong	4	H+12
6		Version	Ulong	4	H+16
7		Station ID	Ulong	4	H+20
8		Message Type	Ulong	4	H+24
9	battery	Is the battery low? 0 = FALSE 1 = TRUE	Enum	4	H+28
10	memory	Is memory low? 0 = FALSE 1 = TRUE	Enum	4	H+32
11	Reserved		Ulong	4	H+36
12	L2	Is L2 enabled? 0 = FALSE 1 = TRUE	Enum	4	H+40
13	Reserved		Ulong	4	H+44
14	epoch	Epoch time (milliseconds)	Ulong	4	H+48
15	motion	Motion state 0 = UNKNOWN 1 = STATIC 2 = KINEMATIC	Ulong	4	H+52
16	Reserved		Ulong	4	H+56
17	rec length	Record length (bytes). The length altogether of the four fields that follow.	Double	8	H+60
18	short ID	Short station ID. A sequence of eight numbers.	Uchar[8]	8	H+68
19	code	COGO code. A sequence of 16 numbers.	Uchar[16]	16	H+76
20	ID length	Long ID length. The length of the long ID field that follows.	Ulong	4	H+92
21	long ID	Long station ID. Variable length. Check Field #20 above.	Uchar[50]	52[a]	H+96
22	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+148
23	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log cmrdatadesca ontime 10 5

ASCII Example:

```
#CMRDATADESCA,COM1,0,76.5,FINESTEERING,1117,162906.461,00100020,b467,399;
2,0,147,39,3,0,2,
FALSE,FALSE,0,TRUE,0,180000,1,0,33,32,32,32,32,99,114,101,102,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,8,85,78,75,78,79,87,78,0*482add29
```

where the bolded 33 in the example above represents the total length of the records that follow:

```
short ID: 32,32,32,32,99,114,101,102, (8 bytes)
COGO code:0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0, (16 bytes)
ID length:8, (1 byte)
long ID:  85,78,75,78,79,87,78,0 (8 bytes)
```

3.4.10 CMRDATAOBS Base Station Satellite Observations RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on CMR standard logs.

Structure: **Message ID: 390**

Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	CMR header	Synch character for the message	Ulong	4	H
3		Message status	Ulong	4	H+4
4		CMR message type	Ulong	4	H+8
5		Message body length	Ulong	4	H+12
6		Version	Ulong	4	H+16
7		Station ID	Ulong	4	H+20
8		Message Type	Ulong	4	H+24
9	#sv	Number of SVs	Ulong	4	H+28
10	epoch	Epoch time (milliseconds)	Ulong	4	H+32
11	clock bias	Is clock bias valid? 0 = NOT VALID 3 = VALID	Ulong	4	H+36
12	clock offset	Clock offset (nanoseconds)	Long	4	H+40
13	# obs	Number of satellite observations with information to follow	Ulong	4	H+44
14	prn	Satellite PRN number	Ulong	4	H+48
15	code flag	Is code P Code? 0 = FALSE 1 = TRUE	Enum	4	H+52
16	L1	Is L1 phase valid? 0 = FALSE 1 = TRUE	Enum	4	H+56
17	L2	Is L2 present? 0 = FALSE 1 = TRUE	Enum	4	H+60
18	L1 psr	L1 pseudorange (1/8 L1 cycles)	Ulong	4	H+64
19	L1 carrier	L1 carrier-code measurement (1/256 L1 cycles)	Long	4	H+68
20	L1 S/N_0	L1 signal-to-noise density ratio	Ulong	4	H+72
21	L1 slip	L1 cycle slip count	Ulong	4	H+76
22	L2 code	Is L2 code available? 0 = FALSE 1 = TRUE	Enum	4	H+80
23	Code type	Is code X-correlation? 0 = FALSE 1 = TRUE	Enum	4	H+84
24	L2 c valid	Is L2 code valid? 0 = FALSE 1 = TRUE	Enum	4	H+88

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
25	L2 p valid	Is L2 phase valid? 0 = FALSE 1 = TRUE	Enum	4	H+92
26	phase full	Is phase full? 0 = FALSE 1 = TRUE	Enum	4	H+96
27	Reserved		Ulong	4	H+100
28	L2 r offset	L2 range offset (1/100 meters)	Long	4	H+104
29	L2 c offset	L2 carrier offset (1/256 cycles)	Long	4	H+108
30	L2 S/N_0	L2 signal-to-noise density ratio	Ulong	4	H+112
31	L2 slip	L2 cycle slip count	Ulong	4	H+116
32...	Next PRN offset = H+48 + (#prns x 72)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log cmrdataobsa ontime 2

ASCII Example:

```
#CMRDATAOBSA,COM1,0,74.0,FINESTEERING,1117,162981.000,00100020,b222,399;
2,0,147,93,3,0,0,
10,21000,3,0,10,
3,FALSE,TRUE,TRUE,8684073,-505,10,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,368,-512,11,1,
15,FALSE,TRUE,TRUE,11936394,129,11,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,270,78,12,1,
18,FALSE,TRUE,TRUE,5334926,186,11,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,164,164,12,1,
21,FALSE,TRUE,TRUE,10590427,-770,10,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,
366,-850,11,1,
17,FALSE,TRUE,TRUE,3262859,32,11,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,325,216,12,1,
26,FALSE,TRUE,TRUE,211264,1213,10,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,390,1069,10,1,
23,FALSE,TRUE,TRUE,8098,209,11,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,265,236,12,1,
28,FALSE,TRUE,TRUE,5090047,-160,6,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,535,-227,9,1,
31,FALSE,TRUE,TRUE,1857322,-1027,7,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,
513,-1063,8,1,
9,FALSE,TRUE,TRUE,51623,-1245,6,1,TRUE,TRUE,TRUE,TRUE,TRUE,0,
599,-1244,9,1*9fe706b0
```

3.4.11 CMRDATAREF Base Station Position RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on CMR standard logs.

Structure: **Message ID: 391**

Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	CMR header	Synch character for the message	Ulong	4	H
3		Message status	Ulong	4	H+4
4		CMR message type	Ulong	4	H+8
5		Message body length	Ulong	4	H+12
6		Version	Ulong	4	H+16
7		Station ID	Ulong	4	H+20
8		Message Type	Ulong	4	H+24
9	battery	Is the battery low? 0 = FALSE 1 = TRUE	Enum	4	H+28
10	memory	Is memory low? 0 = FALSE 1 = TRUE	Enum	4	H+32
11	Reserved		Ulong	4	H+36
12	L2	Is L2 enabled? 0 = FALSE 1 = TRUE	Enum	4	H+40
13	Reserved		Ulong	4	H+44
14	epoch	Epoch time (milliseconds)	Ulong	4	H+48
15	motion	Motion state: 0 = UNKNOWN 1 = STATIC 2 = KINEMATIC	Ulong	4	H+52
16	Reserved		Ulong	4	H+56
17	ECEF-X	Reference ECEF-X position (millimeters)	Double	8	H+60
18	ant hgt	Antenna height (millimeters)	Ulong	4	H+68
19	ECEF-Y	Reference ECEF-Y position (millimeters)	Double	8	H+72
20	e offset	Easting offset (millimeters)	Ulong	4	H+80
21	ECEF-Z	Reference ECEF-Z position (millimeters)	Double	8	H+84
22	n offset	Northing offset (millimeters)	Ulong	4	H+92
23	pos acc	Position accuracy relative to WGS84, see *Table 44, Position Accuracy on Page 138*	Ulong	4	H+96
24	Reserved		Ulong	4	H+100
25	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+104
26	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log cmrdatarefa ontime 10

ASCII Example:

```
#CMRDATAREFA,COM1,0,71.5,FINESTEERING,1117,163185.826,00100020,39f0,399;
2,0,147,25,3,0,1,FALSE,FALSE,0,TRUE,0,216000,1,0,
-1634525357.6077523231506348,0,-3664610762.4997825622558594,
0,166397990,0,15,0*b43b033c
```

Table 44: Position Accuracy

Code	Position Accuracy
0	unknown
1	5 km
2	1 km
3	500 m
4	100 m
5	50 m
6	10 m
7	5 m
8	1 m
9	50 cm
10	10 cm
11	5 cm
12	1 cm
13	5 mm
14	1 mm
15	Exact

3.4.12 COMCONFIG Current COM Port Configuration

This log will output the current COM port configuration for each port on your receiver.

Structure: **Message ID: 317**
Log Type: Polled

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#port	Number of ports with information to follow	Long	4	H
3	port	Serial port identifier, see *Table 13, Serial Port Identifiers on Page 50*	Enum	4	H+4
4	baud	Communication baud rate	Ulong	4	H+8
5	parity	See *Table 14, Parity on Page 50*	Enum	4	H+12
6	databits	Number of data bits	Ulong	4	H+16
7	stopbits	Number of stop bits	Ulong	4	H+20
8	handshake	See *Table 15, Handshaking on Page 50*	Enum	4	H+24
9	echo	When echo is on, the port is transmitting any input characters as they are received. 0 = OFF 1 = ON	Enum	4	H+28
10	breaks	Breaks are turned on or off 0 = OFF 1 = ON	Enum	4	H+32
11	rx type	The status of the receive interface mode, see *Table 24, Serial Port Interface Modes on Page 69*	Enum	4	H+36
12	tx type	The status of the transmit interface mode, *Table 24, Serial Port Interface Modes on Page 69*	Enum	4	H+40
13	response	Responses are turned on or off 0 = OFF 1 = ON	Enum	4	H+44
14	next port offset = H + 4 + (#port x 44)				
15	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4+ (#port x 44)
16	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log comconfiga once

ASCII example:

```
#COMCONFIGA,COM1,0,73.5,FINESTEERING,1114,156455.937,00400000,62b0,33147;
3,
COM1,57600,N,8,1,CTS,OFF,OFF,NOVATEL,NOVATEL,ON,
COM2,9600,N,8,1,N,OFF,ON,NOVATEL,NOVATEL,ON,
COM3,9600,N,8,1,N,OFF,ON,NOVATEL,NOVATEL,ON*dd807b98
```

3.4.13 GPALM Almanac Data NMEA

This log outputs raw almanac data for each satellite PRN contained in the broadcast message. A separate record is logged for each PRN, up to a maximum of 32 records. Following a GPSCard reboot, no records will be output until new broadcast message data is received from a satellite. It takes a minimum of 12.5 minutes to collect a complete almanac following GPSCard boot-up. (The almanac reported here has no relationship to the NovAtel ALMANAC log. Following a cold start, the log will output null fields until a new almanac is collected from a satellite.)

Structure: **Message ID: 217**

Log Type: Asynch

Field	Structure	Field Description	Symbol	Example
1	$GPALM	Log header		$GPALM
2	# msg	Total number of messages logged	x.x	17
3	msg #	Current message number	x.x	17
4	PRN	Satellite PRN number, 01 to 32	xx	28
5	GPS wk	GPS reference week number[a]	x.x	653
6	SV hlth	SV health, bits 17-24 of each almanac page[b]	hh	00
7	ecc	e, eccentricity [c]	hhhh	3EAF
8	alm ref time	toa, almanac reference time [c]	hh	87
9	incl angle	$(sigma)_i$, inclination angle [c]	hhhh	OD68
10	omegadot	OMEGADOT, rate of right ascension [c]	hhhh	FD30
11	rt axis	$(A)^{1/2}$, root of semi-major axis [c]	hhhhhh	A10CAB
12	omega	omega, argument of perigee [c]	hhhhhh	6EE732
13	long asc node	$(OMEGA)_o$,longitude of ascension node [c]	hhhhhh	525880
14	M_O	Mo, mean anomaly [c]	hhhhhh	6DC5A8
15	a_{f0}	af0, clock parameter [c]	hhh	009
16	a_{f1}	af1, clock parameter [c]	hhh	005
17	*xx	Checksum	*hh	*37
18	[CR][LF]	Sentence terminator		[CR][LF]

a Variable length integer, 4-digits maximum from (2) most significant binary bits of Subframe 1, Word 3 reference Table 20-I, ICD-GPS-200, Rev. B, and (8) least significant bits from subframe 5, page 25, word 3 reference Table 20-I, ICD-GPS-200, Rev. B, paragraph 20.3.3.5.1.7[1]

b Reference paragraph 20.3.3.5.1.3, Table 20-VII and Table 20-VIII, ICD-GPS-200, Rev. B

c Reference Table 20-VI, ICD-GPS-200, Rev. B for scaling factors and units.

Example:

```
$GPALM,23,23,191,0000,00,3752,00,ffff,ffff,000000,04cfbe,000011,000000,000,
000,*6c
```

Recommended Input:

log gpalm onchanged

1.To obtain copies of ICD-GPS-200, see ARINC in the appendix on *Standards and References* in *Volume 1* of this manual. See also NMEA contact information here.

3.4.14 GPGGA GPS Fix Data and Undulation NMEA

Time, position and fix-related data of the GPS receiver. For greater precision, but with the loss of the undulation fields, use the GPGGARTK log (see *Page 142*). See also *Table 45, Position Precision of NMEA Logs on Page 143.*

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 218**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGGA	Log header		$GPGGA
2	utc	UTC time of position (hours/minutes/seconds/ decimal seconds)	hhmmss.ss	220147.50
3	lat	Latitude (DDmm.mm)	llll.ll	5106.7194489
4	lat dir	Latitude direction (N = North, S = South)	a	N
5	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.3589020
6	lon dir	Longitude direction (E = East, W = West)	a	W
7	GPS qual	GPS Quality indicator 0 = fix not available or invalid 1 = GPS fix 2 = Differential GPS fix 4 = RTK fixed ambiguity solution 5 = RTK floating ambiguity solution 9 = WAAS [2]	x	1
8	# sats	Number of satellites in use (00-12). May be different to the number in view	xx	08
9	hdop	Horizontal dilution of precision	x.x	0.9
10	alt	Antenna altitude above/below mean sea level (geoid)	x.x	1080.406
11	a-units	Units of antenna altitude (M = meters)	M	M
12	undulation	Undulation	x.x	-16.271
13	u-units	Units of undulation (M = meters)	M	M
14	age	Age of Differential GPS data (in seconds) [1]	xx	,,
15	stn ID	Differential base station ID, 0000-1023	xxxx	,,
16	*xx	Checksum	*hh	*48
17	[CR][LF]	Sentence terminator		[CR][LF]

1 The maximum age reported here is limited to 99 seconds.

2 An indicator of 9 has been temporarily set for WAAS. The NMEA standard for WAAS has not been decided yet.

Example:

```
$GPGGA,144541.00,5106.9819,N,11402.2967,W,1,08,0.9,1067.30,M,-16.27,M,,*65
```

Recommended Input:

log gpgga ontime 1

3.4.15 GPGGARTK Global Position System Fix Data NMEA

Time, position and fix-related data of the GPS receiver. This is output as a GPGGA log but the GPGGARTK log differs from the normal GPGGA log by its extra precision. In order for the position to be output with this extra precision, the undulation fields are unavailable (see the GPGGA log on *Page 141*). See also *Table 45, Position Precision of NMEA Logs on Page 143.*

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 259**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGGA	Log header		$GPGGA
2	utc	UTC time of position (hours/minutes/seconds/ decimal seconds)	hhmmss.ss	220147.50
3	lat	Latitude (DDmm.mm)	llll.ll	5106.7194489
4	lat dir	Latitude direction (N = North, S = South)	a	N
5	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.3589020
6	lon dir	Longitude direction (E = East, W = West)	a	W
7	GPS qual	GPS Quality indicator 0 = fix not available or invalid 1 = GPS fix 2 = Differential GPS fix 4 = RTK fixed ambiguity solution 5 = RTK floating ambiguity solution 9 = WAAS [2]	x	1
8	# sats	Number of satellites in use (00-12). May be different to the number in view	xx	08
9	hdop	Horizontal dilution of precision	x.x	0.9
10	alt	Antenna altitude above/below mean sea level (geoid)	x.x	1080.406
11	units	Units of antenna altitude (M = meters)	M	M
12	null	(This field not available on GPSCards)		,,
13	null	(This field not available on GPSCards)		,,
14	age	Age of Differential GPS data (in seconds) [1]	xx	,,
15	stn ID	Differential base station ID, 0000-1023	xxxx	,,
16	*xx	Checksum	*hh	*48
17	[CR][LF]	Sentence terminator		[CR][LF]

1 The maximum age reported here is limited to 99 seconds.

2 An indicator of 9 has been temporarily set for WAAS. The NMEA standard for WAAS has not been decided yet.

Example:

```
$GPGGA,220147.50,5106.7194489,N,11402.3589020,W,1,08,0.9,1080.406,M,,,,
*48
```

Recommended Input:

log gpggartk ontime 1

3.4.16 GPGLL Geographic Position *NMEA*

Latitude and longitude of present vessel position, time of position fix, and status.

Table 45 compares the position precision of selected NMEA logs.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 219**
Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGLL	Log header		$GPGLL
2	lat	Latitude (DDmm.mm)	llll.ll	5106.7198674
3	lat dir	Latitude direction (N = North, S = South)	a	N
4	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.3587526
5	lon dir	Longitude direction (E = East, W = West)	a	W
6	utc	UTC time of position (hours/minutes/seconds/ decimal seconds)	hhmmss.ss	220152.50
7	data status	Data status: A = Data valid, V = Data invalid	A	A
8	*xx	Checksum	*hh	*1B
9	[CR][LF]	Sentence terminator		[CR][LF]

Example:

```
$GPGLL,5106.9819704,N,11402.2965916,W,144618.00,A*10
```

Recommended Input:

log gpggl ontime 1

Table 45: Position Precision of NMEA Logs

NMEA Log	Latitude (# of decimal places)	Longitude (# of decimal places)	Altitude (# of decimal places)
GPGGA	4	4	2
GPGGARTK	7	7	3
GPGLL	7	7	N/A
GPRMC	7	7	N/A

3.4.17 GPGRS GPS Range Residuals for Each Satellite *NMEA*

Range residuals can be computed in two ways, and this log reports those residuals. Under mode 0, residuals output in this log are used to update the position solution output in the GPGGA message. Under mode 1, the residuals are re-computed after the position solution in the GPGGA message is computed. The GPSCard computes range residuals in mode 1. An integrity process using GPGRS would also require GPGGA (for position fix data), GPGSA (for DOP figures), and GPGSV (for PRN numbers) for comparative purposes.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 220**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGRS	Log header		$GPGRS
2	utc	UTC time of position (hours/minutes/seconds/ decimal seconds)	hhmmss.ss	192911.0
3	mode	Mode 0 =residuals were used to calculate the position given in the matching GGA line (apriori) (not used by GPSCard) Mode 1 =residuals were recomputed after the GGA position was computed (preferred mode)	x	1
4 - 15	res	Range residuals for satellites used in the navigation solution. Order matches order of PRN numbers in GPGSA.	x.x,x.x,.....	-13.8,-1.9,11.4,-33.6,0.9, 6.9,-12.6,0.3,0.6, -22.3
16	*xx	Checksum	*hh	*65
17	[CR][LF]	Sentence terminator		[CR][LF]

Example:

```
$GPGRS,144637.00,1,-1.9,1.1,0.2,-0.1,0.4,0.3,-0.3,-0.7,,,,*45
```

Recommended Input:

log gpgrs ontime 1

✉ If the range residual exceeds ± 99.9, then the decimal part will be dropped. Maximum value for this field is ± 999. The sign of the range residual is determined by the order of parameters used in the calculation as follows:

range residual = calculated range - measured range

3.4.18 GPGSA GPS DOP and Active Satellites *NMEA*

GPS receiver operating mode, satellites used for navigation and DOP values.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 221**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGSA	Log header		$GPGSA
2	mode MA	A = Automatic 2D/3D M = Manual, forced to operate in 2D or 3D	M	M
3	mode 123	Mode: 1 = Fix not available; 2 = 2D; 3 = 3D	x	3
4 - 15	prn	PRN numbers of satellites used in solution (null for unused fields), total of 12 fields	xx,xx,.....	18,03,13,25,16, 24,12,20,,,,
16	pdop	Position dilution of precision	x.x	1.5
17	hdop	Horizontal position and time dilution of precision	x.x	0.9
18	vdop	Vertical dilution of precision	x.x	1.2
19	*xx	Checksum	*hh	*3F
20	[CR][LF]	Sentence terminator		[CR][LF]

Example:

```
$GPGSA,M,3,24,13,05,22,06,30,10,17,,,,,1.6,1.2,1.3*3b
```

Recommended Input:

log gpgsa ontime 1

3.4.19 GPGST Pseudorange Measurement Noise Statistics *NMEA*

Pseudorange measurement noise statistics are translated in the position domain in order to give statistical measures of the quality of the position solution.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 222**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGST	Log header		$GPGST
2	utc	UTC time of position (hours/minutes/seconds/ decimal seconds)	hhmmss.ss	173653.00
3	rms	RMS value of the standard deviation of the range inputs to the navigation process. Range inputs include pseudoranges and DGPS corrections.	x.x	2.73
4	smjr std	Standard deviation of semi-major axis of error ellipse (meters)	x.x	2.55
5	smnr std	Standard deviation of semi-minor axis of error ellipse (meters)	x.x	1.88
6	orient	Orientation of semi-major axis of error ellipse (degrees from true north)	x.x	1.5.2525
7	lat std	Standard deviation of latitude error (meters)	x.x	2.51
8	lon std	Standard deviation of longitude error (meters)	x.x	1.94
9	alt std	Standard deviation of altitude error (meters)	x.x	4.30
10	*xx	Checksum	*hh	*6E
11	[CR][LF]	Sentence terminator		[CR][LF]

Example:

```
$GPGST,173653.00,2.73,2.55,1.88,105.2525,2.51,1.94,4.30*6E
```

Recommended Input:

log gpgst ontime 1

3.4.20 GPGSV GPS Satellites in View NMEA

Number of SVs in view, PRN numbers, elevation, azimuth and SNR value. Four satellites maximum per message. When required, additional satellite data sent in second or third message. Total number of messages being transmitted and the current message being transmitted are indicated in the first two fields.

This log will output null data in all fields until a valid almanac is obtained.

✉ Satellite information may require the transmission of multiple messages. The first field specifies the total number of messages, minimum value 1. The second field identifies the order of this message (message number), minimum value 1.

✉ A variable number of 'PRN-Elevation-Azimuth-SNR' sets are allowed up to a maximum of four sets per message. Null fields are not required for unused sets when less than four sets are transmitted.

Structure: **Message ID: 223**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPGSV	Log header		$GPGSV
2	# msg	Total number of messages, 1 to 3	x	3
3	msg #	Message number, 1 to 3	x	1
4	# sats	Total number of satellites in view	xx	09
5	prn	Satellite PRN number	xx	03
6	elev	Elevation, degrees, 90 maximum	xx	51
7	azimuth	Azimuth, degrees True, 000 to 359	xxx	140
8	SNR	SNR (C/N_0) 00-99 dB, null when not tracking	xx	42
... ...	... ...	Next satellite PRN number, elev, azimuth, SNR, ... Last satellite PRN number, elev, azimuth, SNR,		
variable	*xx	Checksum	*hh	*72
variable	[CR][LF]	Sentence terminator		[CR][LF]

Example:

```
$GPGSV,3,1,11,06,78,273,44,10,54,070,44,17,40,255,40,30,30,206,38*78
$GPGSV,3,2,11,24,21,071,37,13,20,058,37,22,13,323,39,05,07,182,35*73
$GPGSV,3,3,11,26,02,143,,23,02,203,,18,01,014,*46
```

Recommended Input:

log gpgsv ontime 1

3.4.21 GPRMB Navigation Information NMEA

Navigation data from present position to a destination waypoint. The destination is set active by the GPSCard SETNAV command. If SETNAV has been set, a command to log either GPRMB or GPRMC will cause both logs to output data.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 224**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPRMB	Log header		$GPRMB
2	data status	Data status: A = data valid; V = navigation receiver warning	A	V
3	xtrack	Cross track error[1]	x.x	0.011
4	dir	Direction to steer to get back on track (L/R) [2]	a	L
5	origin ID	Origin waypoint ID [3]	c--c	START
6	dest ID	Destination waypoint ID [3]	c--c	END
7	dest lat	Destination waypoint latitude (DDmm.mm [3]	llll.ll	5106.7074000
8	lat dir	Latitude direction (N = North, S = South) [3]	a	N
9	dest lon	Destination waypoint longitude (DDDmm.mm) [3]	yyyyy.yy	11402.349
10	lon dir	Longitude direction (E = East, W = West) [3]	a	E
11	range	Range to destination, nautical miles [4]	x.x	0.0127611
12	bearing	Bearing to destination, degrees True	x.x	153.093
13	vel	Destination closing velocity, knots	x.x	0.3591502
14	arr status	Arrival status: A = perpendicular passed V = destination not reached or passed	A	V
15	*xx	Checksum	*hh	*13
16	[CR][LF]	Sentence terminator		[CR][LF]

1 - If cross track error exceeds 9.99 NM, display 9.99
- Represents track error from intended course
- One nautical mile = 1,852 meters

2 Direction to steer is based on the sign of the crosstrack error, i.e., L = xtrack error (+); R = xtrack error (–)

3 Fields 5, 6, 7, 8, 9, and 10 are tagged from the SETNAV command, see *Page 93*.

4 If range to destination exceeds 999.9 NM, display 999.9

Example:

```
$GPRMB,A,0.00,L,start,end,5224.0000000,N,11500.0000000,W,85.0,335.4,0.0,V*13
```

Recommended Input:

```
log gprmb ontime 1
```

3.4.22 GPRMC GPS Specific Information NMEA

Time, date, position, track made good and speed data provided by the GPS navigation receiver. RMC and RMB are the recommended minimum navigation data to be provided by a GPS receiver.

A comparison of the position precision between this log and other selected NMEA logs can be seen in *Table 45, Position Precision of NMEA Logs on Page 143*.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 225**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPRMC	Log header		$GPRMC
2	utc	UTC of position	hhmmss.ss	220216.50
3	pos status	Position status: A = data valid V = data invalid	A	A
4	lat	Latitude (DDmm.mm)	llll.ll	5106.7187663
5	lat dir	Latitude direction (N = North, S = South)	a	N
6	lon	Longitude (DDDmm.mm)	yyyyy.yy	11402.3581636
7	lon dir	Longitude direction (E = East, W = West)	a	W
8	speed Kn	Speed over ground, knots	x.x	0.3886308
9	track true	Track made good, degrees True	x.x	130.632
10	date	Date: dd/mm/yy	xxxxxx	150792
11	mag var	Magnetic variation, degrees [2]	x.x	0.000
12	var dir	Magnetic variation direction E/W [1]	a	E
13	*xx	Checksum	*hh	*4B
14	[CR][LF]	Sentence terminator		[CR][LF]

1 Easterly variation (E) subtracts from True course
Westerly variation (W) adds to True course

2 Note that this field is the actual magnetic variation East or West and is the inverse sign of the value entered into the *MAGVAR* command, see *Page 74* for more information.

Example:

```
$GPRMC,144756.00,A,5106.9821384,N,11402.2962485,W,0.023,309.6,040800,0.0,E*41
```

Recommended Input:

```
log gprmc ontime 1
```

3.4.23 GPSEPHEM Decoded GPS Ephemerides

A single set of GPS ephemeris parameters.

Structure: **Message ID: 7**
Log Type: Asynch

Field#	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	PRN	Satellite PRN number	Ulong	4	H
3	tow	Time stamp of subframe 0 (seconds)	Double	8	H+4
4	health	Health status. A 6-bit health code as defined in ICD-GPS-200[a].	Ulong	4	H+12
5	IODE1	Issue of ephemeris data 1	Ulong	4	H+16
6	IODE2	Issue of ephemeris data 2	Ulong	4	H+20
7	week	GPS week number	Ulong	4	H+24
8	z week	Z count week number. This is the week number from subframe 1 of the ephemeris. The 'toe week' (field #7) is derived from this to account for rollover.	Ulong	4	H+28
9	toe	Reference time for ephemeris, seconds	Double	8	H+32
10	A	Semi-major axis, meters	Double	8	H+40
11	ΔN	Mean motion difference, radians/second	Double	8	H+48
12	M_0	Mean anomaly of reference time, radians	Double	8	H+56
13	ecc	Eccentricity, dimensionless	Double	8	H+64
14	ω	Argument of perigee, radians	Double	8	H+72
15	cuc	Argument of latitude (amplitude of cosine, radians)	Double	8	H+80
16	cus	Argument of latitude (amplitude of sine, radians)	Double	8	H+88
17	crc	Orbit radius (amplitude of cosine, meters)	Double	8	H+96
18	crs	Orbit radius (amplitude of sine, meters)	Double	8	H+104
19	cic	Inclination (amplitude of cosine, radians)	Double	8	H+112
20	cis	Inclination (amplitude of sine, radians)	Double	8	H+120
21	I_0	Inclination angle at reference time, radians	Double	8	H+128
22	$\dot{I}$	Rate of inclination angle, radians/second	Double	8	H+136
23	ω_0	Right ascension, radians	Double	8	H+144
24	$\dot{\omega}$	Rate of right ascension, radians/second	Double	8	H+152
25	iodc	Issue of data clock	Ulong	4	H+160
26	toc	SV clock correction term, seconds	Double	8	H+164
27	tgd	Estimated group delay difference, seconds	Double	8	H+172
28	a_{f0}	Clock aging parameter, seconds (s)	Double	8	H+180
29	a_{f1}	Clock aging parameter, (s/s)	Double	8	H+188
30	a_{f2}	Clock aging parameter, (s/s/s)	Double	8	H+196
31	AS	Anti-spoofing on: 0 = FALSE 1 = TRUE	Enum	4	H+204
32	N	Corrected mean motion, radians/second	Double	8	H+208

Field#	Field type	Data Description	Format	Binary Bytes	Binary Offset
33	URA	User Range Accuracy variance, m^2. The ICD[a] specifies that the URA index transmitted in the ephemerides can be converted to a nominal standard deviation value using an algorithm listed there. We publish the square of the nominal value (variance). The correspondance between the original URA index and the value output is shown in *Table 46*.	Double	8	H+216
34	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+224
35	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. To obtain copies of ICD-GPS-200, see ARINC in the appendix on *Standards and References* in *Volume 1* of this manual.

Table 46: URA Variance

Index Value (m)	A: Standard Deviations (m)	Variance: A^2 (m^2)
0	2.0	4
1	2.8	7.84
2	4.0	16
3	5.7	32.49
4	8	56
5	11.3	127.69
6	16.0	256
7	32.0	1024
8	64.0	4096
9	128.0	16384
10	256.0	65536
11	512.0	262144
12	1024.0	1048576
13	2048.0	4194304
14	4096.0	16777216
15	8192.0	67108864

Recommended Input:

log gpsephema onchanged

ASCII Example:

```
#GPSEPHEMA,COM1,6,52.5,SATTIME,1048,505920.000,00000028,e05d,0;
7,505920.0,0,61,61,1048,1048,511200.0,2.65611412e+07,5.11057002e-09,
-2.44168504e+00,1.081193e-02,-2.09812298e+00,1.81607902e-06,
4.31202352e-06,2.90031250e+02,3.44062500e+01,1.73225999e-07,
-5.40167093e-08,9.51914192e-01,8.32177521e-11,2.763202e+00,
-8.406064e-09,61,511200.0,-1.86264515e-09,3.9749e-04,4.8885e-11,
0.00000,TRUE,1.45852555e-04,1.02400000e+03*0491840f
…
#GPSEPHEMA,COM1,0,52.5,SATTIME,1048,505950.000,00000028,e05d,0;
4,505950.0,0,171,171,1048,1048,511184.0,2.65606308e+07,4.19731769e-09,
1.07102804e+00,5.387515e-03,-6.16884846e-01,-2.86661088e-06,
7.91251659e-06,2.31062500e+02,-5.39687500e+01,1.34110451e-07,
-1.49011612e-08,9.77176578e-01,-3.72872675e-10,-2.430120e+00,
-8.025334e-09,171,511184.0,-6.05359674e-09,7.0939e-04,2.0009e-11,
0.00000,TRUE,1.45855846e-04,1.02400000e+03*a6beec64
```

3.4.24 GPVTG Track Made Good And Ground Speed NMEA

The track made good and speed relative to the ground.

This log will output null data in all fields until a valid almanac is obtained.

Structure: **Message ID: 226**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPVTG	Log header		$GPVTG
2	track true	Track made good, degrees True	x.x	24.168
3	T	True track indicator	T	T
4	track mag	Track made good, degrees Magnetic; Track mag = Track true + (MAGVAR correction) See the *MAGVAR* command, *Page 74*.	x.x	24.168
5	M	Magnetic track indicator	M	M
6	speed Kn	Speed over ground, knots	x.x	0.4220347
7	N	Nautical speed indicator (N = Knots)	N	N
8	speed Km	Speed, kilometers/hour	x.x	0.781608
9	K	Speed indicator (K = km/hr)	K	K
10	*xx	Checksum	*hh	*7A
11	[CR][LF]	Sentence terminator		[CR][LF]

Example:

```
$GPVTG,337.817,T,337.817,M,0.020,N,0.042,K*4a
```

Recommended Input:

log gpvtg ontime 1

3.4.25 GPZDA UTC Time and Date *NMEA*

This log will output null data in all fields until a valid almanac is downloaded from a satellite. Any alternate almanac already in NVM will not be output.

Structure: **Message ID: 227**

Log Type: Synch

Field	Structure	Field Description	Symbol	Example
1	$GPZDA	Log header		$GPZDA
2	utc	UTC time	hhmmss.ss	220238.00
3	day	Day, 01 to 31	xx	15
4	month	Month, 01 to 12	xx	07
5	year	Year	xxxx	1992
6	null	Local zone description - not available	xx	, ,
7	null	Local zone minutes description - not available [1]	xx	, ,
8	*xx	Checksum	*hh	*6F
9	[CR][LF]	Sentence terminator		[CR][LF]

1 Local time zones are not supported by the GPSCard. Fields 6 and 7 will always be null.

Example:

```
$GPZDA,145129.00,04,08,2000,,*62
```

Recommended Input:

log gpzda ontime 1

3.4.26 IONUTC Ionospheric and UTC Data

The Ionospheric Model parameters (ION) and the UTC Time parameters (UTC) are provided.

Structure: **Message ID: 8**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	a0	Alpha parameter constant term	Double	8	H
3	a1	Alpha parameter 1st order term	Double	8	H+8
4	a2	Alpha parameter 2nd order term	Double	8	H+16
5	a3	Alpha parameter 3rd order term	Double	8	H+24
6	b0	Beta parameter constant term	Double	8	H+32
7	b1	Beta parameter 1st order term	Double	8	H+40
8	b2	Beta parameter 2nd order term	Double	8	H+48
9	b3	Beta parameter 3rd order term	Double	8	H+56
10	utc wn	UTC reference week number	Ulong	4	H+64
11	tot	Reference time of UTC parameters	Ulong	4	H+68
12	a0	2nd alpha parameter constant term	Double	8	H+72
13	a1	2nd alpha parameter 1st order term	Double	8	H+80
14	wn lsf	Future week number	Ulong	4	H+88
15	dn	Day number	Ulong	4	H+92
16	deltat ls	Delta time due to leap seconds	Long	4	H+96
17	deltat lsf	Future delta time due to leap seconds	Long	4	H+100
18	Reserved		Ulong	4	H+104
19	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+108
20	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log ionutca onchanged

ASCII Example:

```
#IONUTCA,COM1,0,56.0,FINESTEERING,1048,496800.089,00000128,3d19,0;
2.7939677238464355e-08,1.4901161193847654e-08,-1.1920928955078123e-07,
5.9604644775390616e-08,1.4131200000000000e+05,
-8.1920000000000000e+04,6.5536000000000000e+04,
-4.5875200000000000e+05,1049,61440,-7.450580596923828e-09,
0.000000000,990,5,13,13,0*3b80134b
```

3.4.27 LOGLIST List of System Logs

Outputs a list of log entries in the system. *Table 47* is a table of the binary output while *Table 48* shows the ASCII output. See also the RXCONFIG log on *Page 218* for a list of current command settings.

Structure: **Message ID: 5**
Log Type: Polled

Table 47: LOGLIST Binary Format

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#logs	Number of messages to follow, maximum = 20	Long	4	H
3	port	Output port, see *Table 5, Serial Port Identifiers on Page 18*	Enum	4	H+4
4	message	Message ID of log	Ushort	2	H+8
5	message type	Bits 0-4 = Reserved Bits 5-6 = Format 00 = Binary 01 = ASCII 10 = Abbreviated ASCII, NMEA 11 = Reserved Bit 7 = Response Bit (see *Section 1.2, Responses on Page 19*) 0 = Original Message 1 = Response Message	Char	1	H+10
6	reserved		Char	3[a]	H+11
7	trigger	0 = ONNEW 1 = ONCHANGED 2 = ONTIME 3 = ONNEXT 4 = ONCE	Enum	4	H+14
8	period	Log period for ONTIME	Double	8	H+18
9	offset	Offset for period (ONTIME trigger)	Double	8	H+26
10	hold	0 = NOHOLD 1 = HOLD	Enum	4	H+34
11...	Next log offset = H + 4 + (#logs x 34)				
variable	xxxx	32-bit CRC	Hex	4	H+4+ (#logs x 34)

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment

Table 48: LOGLIST ASCII Format

Field #	Field type	Data Description	Format
1	header	Log header	
2	#port	Number of messages to follow, maximum = 20	Long
3	port	Output port, see *Table 5, Serial Port Identifiers on Page 18*	Enum
4	message	Message name of log with no suffix for abbreviated ascii, an A suffix for ascii and a B suffix for binary.	Char [ ]
5	trigger	ONNEW ONCHANGED ONTIME ONNEXT ONCE	Enum
6	period	Log period for ONTIME	Double
7	offset	Offset for period (ONTIME trigger)	Double
8	hold	NOHOLD HOLD	Enum
9...	Next port		
variable	xxxx	32-bit CRC	Hex
variable	[CR][LF]	Sentence terminator	-

Recommended Input:

log loglista once

ASCII Example:

```
#LOGLISTA,COM1,0,59.5,FINESTEERING,1037,320631.415,00000028,b12b,0;
 6,
     COM1,RXSTATUSEVENTA,ONCHANGED,0.000000,0.000000,HOLD,
     COM2,RXSTATUSEVENTA,ONCHANGED,0.000000,0.000000,HOLD,
     COM3,RXSTATUSEVENTA,ONCHANGED,0.000000,0.000000,HOLD,
     COM1,AVEPOS,ONCE,0.000000,0.000000,NOHOLD,
     COM1,AVEPOSA,ONCE,0.000000,0.000000,NOHOLD,
     COM1,LOGLISTA,ONCE,0.000000,0.000000,NOHOLD*0d6bb570
```

3.4.28 MARKPOS Position at Time of MarkIn Event

This log contains the estimated position of the antenna at detected mark impulse. It uses the last valid position and velocities to extrapolate the position at time of mark. Refer to the *Technical Specifications* appendix in *Volume 1* of this manual for Mark Input pulse specifications. The latched time of mark impulse is in GPS weeks and seconds into the week. The resolution of the latched time is 49 ns. See also the notes on MARKPOS in the MARKTIME log on *Page 158*.

Structure: **Message ID: 181**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status (see *Table 40, Solution Status on Page 124*)	Enum	4	H
3	pos type	Position type (see *Table 39, Position or Velocity Type on Page 123*)	Enum	4	H+4
4	lat	Latitude	Double	8	H+8
5	lon	Longitude	Double	8	H+16
6	hgt	Height above mean sea level	Double	8	H+24
7	undulation	Undulation	Float	4	H+32
8	datum id#	Datum ID number (see *Chapter 2, Table 17, Datum Transformation Parameters on Page 54*)	Enum	4	H+36
9	lat σ	Latitude standard deviation	Float	4	H+40
10	lon σ	Longitude standard deviation	Float	4	H+44
11	hgt σ	Height standard deviation	Float	4	H+48
12	stn id	Base station ID	Char[4]	4	H+52
13	diff_age	Differential age	Float	4	H+56
14	sol_age	Solution age in seconds	Float	4	H+60
15	#obs	Number of observations tracked	Uchar	1	H+64
16	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+65
17	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+66
18	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+67
19	Reserved		Uchar	1	H+68
20			Uchar	1	H+69
21			Uchar	1	H+70
22			Uchar	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log markposa onmark

Example:

```
#MARKPOSA,COM1,0,63.0,COARSE,1027,322788.595,00000000,6221,0;
SOL_COMPUTED,WIDELANE,51.11227014,-114.03907552,1003.799,-16.199,61,
7.793,3.223,34.509,"",0.0,60.000,8,4,4,4,0,0,0,0,0,1*5603d9f8
```

3.4.29 MARKTIME Time of MarkIn Event

This log contains the time of the detected Mark Input pulse leading edge as detected at the Mark Input I/O port. The resolution of this measurement is 49ns. Refer to the *Technical Specifications* appendix in *Volume 1* of this manual for the Mark Input pulse specifications.

Structure: **Message ID: 231**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	week	GPS week number	Long	4	H
3	seconds	Seconds into the week as measured from the receiver clock, coincident with the time of electrical closure on the Mark Input port.	Double	8	H+4
4	offset	Receiver clock offset, in seconds. A positive offset implies that the receiver clock is ahead of GPS Time. To derive GPS time, use the following formula: GPS time = receiver time - (offset)	Double	8	H+12
5	offset std	Standard deviation of receiver clock offset, in seconds	Double	8	H+20
6	utc offset	This field represents the offset of GPS time from UTC time, computed using almanac parameters. UTC time is GPS time plus the current UTC offset plus the receiver clock offset. UTC time = GPS time + offset + UTC offset[a]	Double	8	H+28
7	status	Clock model status, see *Table 41, Clock Model Status on Page 128*	Enum	4	H+36
8	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+40
9	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. 0 indicates that UTC time is unknown because there is no almanac available in order to acquire the UTC offset.

Recommended Input:

```
log marktimea onmark
```

Example:

```
#MARKTIME,COM1,0,63.0,COARSE,1027,322788.595,00000000,6221,0;
653,338214.773382376,0.000504070,0.000000013,-8.000000000,0 *523581ba
```

✉ Using the ‘onmark’ trigger with this, or the MARKPOS log, is no different than using the ‘onnew’ trigger.

✉ Only the MARKPOS log, the MARKTIME log, and ‘polled’ log types are generated ‘on the fly’ at the exact time of the mark. Synchronous and asynchronous logs will output the most recently available data.

3.4.30 MATCHEDPOS Matched RTK Position RTK

This log represents positions that have been computed from time matched base and rover observations. There is no base station extrapolation error on these positions because they are based on buffered measurements; they lag real time by some amount depending on the latency of the data link. If the rover receiver has not been enabled to accept RTK differential data, or is not actually receiving data leading to a valid solution, this will be reflected by the code shown in field #2 (solution status) and #3 (position type).

This log provides the best accuracy in static operation. For lower latency in kinematic operation, see the RTKPOS or BESTPOS logs.

The data in the logs will change only when a base observation (RTCM, RTCA, or CMR) changes.

A good message trigger for this log is "ONCHANGED". Then, only positions related to unique base station messages will be produced, and the existence of this log will indicate a successful link to the base station.

Structure: **Message ID: 96**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status (see *Table 40, Solution Status on Page 124*)	Enum	4	H
3	pos type	Position type (see *Table 39, Position or Velocity Type on Page 123*)	Enum	4	H+4
4	lat	Latitude	Double	8	H+8
5	lon	Longitude	Double	8	H+16
6	hgt	Height above mean sea level	Double	8	H+24
7	undulation	Undulation	Float	4	H+32
8	datum id#	Datum ID number (see *Chapter 2, Table 17, Datum Transformation Parameters on Page 54*)	Enum	4	H+36
9	lat σ	Latitude standard deviation	Float	4	H+40
10	lon σ	Longitude standard deviation	Float	4	H+44
11	hgt σ	Height standard deviation	Float	4	H+48
12	stn id	Base station ID	Char[4]	4	H+52
13	Reserved		Float	4	H+56
14			Float	4	H+60
15	#obs	Number of observations tracked	Uchar	1	H+64
16	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+65
17	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+66
18	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+67

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
19	Reserved		Uchar	1	H+68
20			Uchar	1	H+69
21			Uchar	1	H+70
22			Uchar	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log matchedposa onchanged

ASCII Example:

```
#MATCHEDPOSA,COM1,0,39.0,FINESTEERING,1037,507654.000,00000028,abce,0;
SOL_COMPUTED,L1_FLOAT,51.11638133582,-114.03829215909,1051.6231,
-16.271240054,61,0.0619,0.0870,0.0531,"0",0.0,60.000,10,10,10,
0,0,0,0,0*5a475478
```

3.4.31 MATCHEDXYZ Matched RTK Cartesian Position RTK

This log contains the receiver's matched position in ECEF coordinates. It represents positions that have been computed from time matched base and rover observations. There is no base station extrapolation error on these positions because they are based on buffered measurements; they lag real time by some amount depending on the latency of the data link. If the rover receiver has not been enabled to accept RTK differential data, or is not actually receiving data leading to a valid solution, this will be reflected by the code shown in field #2 (solution status) and #3 (position type). See *Figure 6, Page 127* for a definition of the ECEF coordinates.

This log provides the best accuracy in static operation. For lower latency in kinematic operation, see the RTKXYZ or BESTXYZ logs on *Pages 126 and 216* respectively.The data in the logs will change only when a base observation (RTCM, RTCA, or CMR) changes.

A good message trigger for this log is "ONCHANGED". Then, only positions related to unique base station messages will be produced, and the existence of this log will indicate a successful link to the base station.

The time stamp in the header is the time of the matched observations that the computed position is based on, not the current time.

Structure: **Message ID: 242**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	P-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	P-X	Position X-coordinate (m)	Double	8	H+8
5	P-Y	Position Y-coordinate (m)	Double	8	H+16
6	P-Z	Position Z-coordinate (m)	Double	8	H+24
7	P-X σ	Standard deviation of P-X (m)	Float	4	H+32
8	P-Y σ	Standard deviation of P-Y (m)	Float	4	H+36
9	P-Z σ	Standard deviation of P-Z (m)	Float	4	H+40
18	stn ID	Base station ID	Char[4]	4	H+44
22	#obs	Number of observations tracked	Uchar	1	H+48
23	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+49
24	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+50
25	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+51
26	Reserved		Char	1	H+52
27			Char	1	H+53
28			Char	1	H+54
29			Char	1	H+55
30	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+56
31	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log matchedxyza onchanged

ASCII Example:

```
#MATCHEDXYZA,COM1,0,55.5,FINESTEERING,1058,410372.000,00000028,7eb2,37;
SOL_COMPUTED,NARROW_INT,-1634527.499,-3664605.328,4942486.979,0.0028,0.0049,
0.0047,"AAAA",9,6,6,6,0,0,0,0*17085764
```

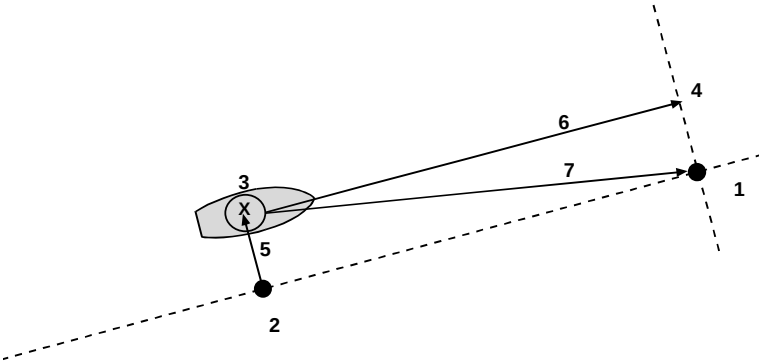
3.4.32 NAVIGATE User Navigation Data

This log reports the status of the waypoint navigation progress. It is used in conjunction with the SETNAV command, see *Page 93*.

See *Figure 7, below,* for an illustration of navigation parameters.

✉ The SETNAV command must be enabled before valid data will be reported from this log.

Figure 7: Navigation Parameters



Reference	Description
1	TO lat-lon
2	X-Track perpendicular reference point
3	Current GPS position
4	A-Track perpendicular reference point
5	X-Track (cross track)
6	A-Track (along track)
7	Distance and bearing from 3 to 1

Table 49: Navigation Data Type

Navigation Data Type		Description
Binary	ASCII	
0	GOOD	Navigation is good
1	NOVELOCITY	Navigation has no velocity
2	BADNAV	Navigation calculation failed for an unknown reason
3	FROM_TO_SAME	“From” is too close to “To” for computation
4	TOO_CLOSE_TO_TO	Position is too close to “To” for computation
5	ANTIPODAL_WAYPTS	Waypoints are antipodal on surface

Structure: **Message ID: 161**
Log Type: Synch

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+8
5	nav type	Navigation data type (see *Table 49, Navigation Data Type on Page 162*).	Enum	4	H+12
6	distance	Straight line horizontal distance from current position to the destination waypoint, in meters (see *Figure 7, Navigation Parameters on Page 162).* This value is positive when approaching the waypoint and becomes negative on passing the waypoint.	Double	8	H+16
7	bearing	Direction from the current position to the destination waypoint in degrees with respect to True North (or Magnetic if corrected for magnetic variation by MAGVAR command)	Double	8	H+24
8	along track	Horizontal track distance from the current position to the closest point on the waypoint arrival perpendicular; expressed in meters. This value is positive when approaching the waypoint and becomes negative on passing the waypoint.	Double	8	H+32
9	xtrack	The horizontal distance (perpendicular track-error) from the vessel's present position to the closest point on the great circle line that joins the FROM and TO waypoints. If a "track offset" has been entered in the SETNAV command, xtrack will be the perpendicular error from the "offset track". Xtrack is expressed in meters. Positive values indicate the current position is right of the Track, while negative offset values indicate left.	Double	8	H+40
10	eta week	Estimated GPS week number at time of arrival at the "TO" waypoint along track arrival perpendicular based on current position and speed, in units of GPS weeks. If the receiving antenna is moving at a speed of less than 0.1 m/sec in the direction of the destination, the value in this field will be "9999".	Ulong	4	H+48
11	eta secs	Estimated GPS seconds into week at time of arrival at destination waypoint along track arrival perpendicular, based on current position and speed, in units of GPS seconds into the week. If the receiving antenna is moving at a speed of less than 0.1 m/sec in the direction of the destination, the value in this field will be "0.000".	Double	8	H+52
12	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+60
13	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log navigatea ontime 1

ASCII Example:

```
#NAVIGATEA,COM1,0,61.5,FINESTEERING,1049,247860.000,00000128,74a8,0;
SOL_COMPUTED,SINGLE,DOPPLER_VELOCITY,FROM_TO_SAME,0.0000,0.000000,0.0000,0.00
00,0,0.000*eee963ff
```

3.4.33 NMEA Standard Logs

See the chapter on *Message Formats* in *Volume 1* of this manual for more detail on the structure of NMEA logs. The details for the following NMEA logs are listed alphabetically in this chapter.

GPALM	**ALMANAC DATA**
GPGGARTK	**GLOBAL POSITION SYSTEM FIX DATA**
GPGGA	**GLOBAL POSITION SYSTEM FIX DATA AND UNDULATION**
GPGLL	**GEOGRAPHIC POSITION**
GPGRS	**GPS RANGE RESIDUALS FOR EACH SATELLITE**
GPGSA	**GPS DOP AN ACTIVE SATELLITES**
GPGST	**PSEUDORANGE MEASUREMENT NOISE STATISTICS**
GPGSV	**GPS SATELLITES IN VIEW**
GPRMB	**NAVIGATION INFORMATION**
GPRMC	**GPS SPECIFIC INFORMATION**
GPVTG	**TRACK MADE GOOD AND GROUND SPEED**
GPZDA	**UTC TIME AND DATE**

3.4.34 OMNIHPPOS OMNISTAR HP Position

Outputs OmniSTAR High Precision (HP) position information.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Structure: **Message ID: 495**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	lat	Latitude	Double	8	H+8
5	lon	Longitude	Double	8	H+16
6	hgt	Height above mean sea level	Double	8	H+24
7	undulation	Undulation	Float	4	H+32
8	datum id#	Datum ID number (see *Chapter 2, Table 17, Datum Transformation Parameters on Page 54*)	Enum	4	H+36
9	lat σ	Latitude standard deviation	Float	4	H+40
10	lon σ	Longitude standard deviation	Float	4	H+44
11	hgt σ	Height standard deviation	Float	4	H+48
12	stn id	Base station ID	Char[4]	4	H+52
13	diff_age	Differential age	Float	4	H+56
14	sol_age	Solution age in seconds	Float	4	H+60
15	#obs	Number of observations tracked	Uchar	1	H+64
16	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+65
17	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+66
18	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+67
19	Reserved		Uchar	1	H+68
20			Uchar	1	H+69
21			Uchar	1	H+70
22			Uchar	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log omnihpposa ontime 1

ASCII Example:

```
#OMNIHPPOSA,COM1,0,72.0,FINESTEERING,1161,321910.000,00000000,ad26,683;
SOL_COMPUTED,OMNISTAR_HP,51.11635244839,-114.03819232612,1064.1015,-16.2713,
WGS84,0.1371,0.1390,0.2741,"",5.000,0.000,7,6,6,6,0,0,0,0*66c318fb
```

3.4.35 OMNIINFO OmniSTAR Configuration Information

This log outputs configuration information for an OmniSTAR VBS (virtual base station) and HP (high precision) service.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Structure: **Message ID: 509**
Log Type: Sync

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	freq	Selected frequency for OmniSTAR service (kHz)	Ulong	4	H
3	baud	Communication baud rate from OmniSTAR satellite	Ulong	4	H+4
4	ID	OmniSTAR signal service ID	Ushort	2	H+8
5	Reserved		Ushort	2	H+10
6	OSN	OmniSTAR serial number	Ulong	4	H+12
7	subscrip	OmnSTAR VBS subscription type (see *Table 50 on Page 167*)	Enum	4	H+16
8	exp week	GPS week number of OmniSTAR VBS expiration date [a]	Ulong	4	H+20
9	exp secs	Number of seconds into the GPS week of OmniSTAR VBS expiration date [a]	Ulong	4	H+24
10	Reserved		Enum	4	H+28
11			Ulong	4	H+32
12			Ulong	4	H+36
13			Ulong	4	H+40
14	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
15	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. If the subscription type is COUNTDOWN, see Field #7 above, the expiration week and expiration seconds into the GPS week will contain the amount of VBS running time remaining in the subscription.
If the subscription type is COUNTDOWNOVERRUN, the expiration week and expiration seconds into GPS week will count the amount of the overrun time.

Table 50: OmniSTAR Subscription Type

Binary	ASCII	Description
0	EXPIRED	The OmniSTAR VBS subscription has expired or does not exist.
1	FIXEDTIME	The OmniSTAR VBS subscription will expire at a fixed date and time.
2	COUNTDOWN	The OmniSTAR VBS subscription will expire after the specified amount of running time.
3	COUNTDOWNOVERRUN	The COUNTDOWN subscription has expired but has entered a brief grace period. Resubscribe immediately.

Recommended Input:

log omniinfoa ontime 1

ASCII Example:

```
#OMNIINFOA,COM1,0,64.5,FINESTEERING,1164,240223.642,00000000,4797,33477;
1551489,1200,c685,0,704312,EXPIRED,0,0,NONE,0,0,0*e8bea6a3
```

3.4.36 OMNISTAT OmniSTAR Status Information

This log outputs status information for an OmniSTAR VBS (virtual base station) and HP (high precision) service.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Structure: **Message ID: 510**
Log Type: Synch

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	freq	Measured frequency of OmniSTAR signal (kHz).	Ulong	4	H
3	C/N_0	Carrier to noise density ratio $C/N_0 = 10[\log_{10}(S/N_0)]$ (dB-Hz)	Float	4	H+4
4	locktime	Number of seconds of continuous tracking (no cycle slipping).	Float	4	H+8
5	Reserved		Float	4	H+12
6	tracking	Tracking status of OmniSTAR signal	Hex	2	H+16
7	status	Status word from the VBS process.	Hex	2	H+20
8	#bytes	Number of bytes fed to the VBS process.	Ulong	4	H+24
9	#good dgps	Number of VBS updates.	Ulong	4	H+28
10	#bad data	Number of missing VBS updates.	Ulong	4	H+32
11	Reserved		Ushort	2	H+36
12			Ushort	2	H+40
13			Ulong	4	H+44
14			Ulong	4	H+48
15			Ulong	4	H+52
16	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+56
17	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log omnistata ontime 1

ASCII Example:

```
#OMNISTATA,COM1,0,64.0,FINESTEERING,1164,240276.647,00000000,a578,33477;
1551488896,41.99,149.7,0.00,0082,0000,18742,33,0,0,0,0,0,0*634d507a
```

Table 51: OmniSTAR Signal Tracking Status

Nibble #	Bit #	Mask	Description	Range Value
N0	0	0x0001	Tracking State	0 = Searching, 1 = Pull-in, 2 = Tracking
	1	0x0002		
	2	0x0004	Reserved	
	3	0x0008		
N1	4	0x0010		
	5	0x0020		
	6	0x0040		
	7	0x0080	Phase Locked	0 = Not Locked, 1 = Locked
N2	8	0x0100	Reserved	
	9	0x0200		
	10	0x0400		
	11	0x0800		
N3	12	0x1000		
	13	0x2000		
	14	0x4000		
	15	0x8000	Error	0 = Good, 1 = Error

Table 52: VBS Status Word

Nibble #	Bit #	Mask	Description	Bit = 0	Bit = 1
N0	0	0x0001	Subscription Expired[a]	True	False
	1	0x0002	Out of Region[a]	True	False
	2	0x0004	Wet Error[a]	True	False
	3	0x0008	Link Error[a]	True	False
N1	4	0x0010	No Remote Sites	True	False
	5	0x0020	No Almanac	True	False
	6	0x0040	No Position	True	False
	7	0x0080	No Time	True	False
N2	8	0x0100	Reserved		
	9	0x0200			
	10	0x0400			
	11	0x0800			
N3	12	0x1000			
	13	0x2000			
	14	0x4000			
	15	0x8000	Updating Data	True	False

a. Contact OmniSTAR for subscription support. All other status values will be updated by collecting the OmniSTAR data for 20-35 minutes.

3.4.37 PASSCOM1, PASSCOM2, PASSCOM3 Redirect Data

There are three pass-through logs PASSCOM1, PASSCOM2 and PASSCOM3. The pass-through logging feature enables the receiver to redirect any ascii or binary data that is input at a specified port (COM1, COM2 or COM3) to any specified receiver port (COM1, COM2 or COM3). Data can be passed through a disabled port, see the *INTERFACEMODE* command on *Page 69*, and be output from an alternative port using the pass-through logs PASSCOM1, PASSCOM2 and PASSCOM3. It also allows the receiver to perform bi-directional communications with other devices such as a modem, terminal or another receiver.

If the data being injected is ASCII then the data will be grouped together with the following rules:

- blocks of 80 characters
- any block of characters ending in a <CR>
- any block of characters ending in a <LF>
- any block remaining in the receiver code when a time-out occurs

If the data being injected is binary then the data will be grouped as follows:

- blocks of 80 bytes
- any block remaining in the receiver code when a time-out occurs

If a binary value is encountered in an ASCII output, then the byte is output as a hexadecimal byte preceded by a back slash and an x. For example 0a is output as \x0a. An actual '\' in the data is output as '\\'.The output counts as one pass-through byte although it is four characters.

For more information on pass-through logging, please see the *Operation* chapter in *Volume 1* of this manual.

Structure: **PASSCOM1 Message ID: 233**
PASSCOM2 Message ID: 234
PASSCOM3 Message ID: 235
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#bytes	Number of bytes to follow	Ulong	4	H
3	data	Message data	Char [80]	80	H+4
4	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+8+(#bytes)
5	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log passcom1a onnew

ASCII Example:

```
#PASSCOM3A,COM1,0,69.5,FINESTEERING,1060,161079.486,00000000,386c,40;
28,log com3 bestposa ontime 2 \x0d\x0a*8ad22436
#PASSCOM3A,COM1,0,68.5,FINESTEERING,1060,161083.617,00000000,386c,40;
10,unlogall\x0d\x0a*e061bbad
```

3.4.38 PORTSTATS COM Port Statistics

This log conveys various status parameters of the receiver COM ports.

The GPSCard maintains a running count of a variety of status indicators of the data link. This log outputs a report of those indicators.

Parity and framing errors will occur if poor transmission lines are encountered or if there is an incompatibility in the data protocol. If errors occur, you may need to confirm the bit rate, number of data bits, number of stop bits and parity of both the transmit and receiving ends. Characters may be dropped when the CPU is overloaded.

Structure: **Message ID: 72**
Log Type: Polled

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#port	Number of ports with information to follow	Long	4	H
3	port	Serial port identifier, see *Table 13, Serial Port Identifiers on Page 50*	Enum	4	H+4
4	rx chars	Total number of characters received through this port	Ulong	4	H+8
5	tx chars	Total number of characters transmitted through this port	Ulong	4	H+12
6	acc rx chars	Total number of accepted characters received through this port	Ulong	4	H+16
7	dropped chars	Number of software overruns	Ulong	4	H+20
8	interrupts	Number of interrupts on this port	Ulong	4	H+24
9	breaks	Number of breaks	Ulong	4	H+28
10	par err	Number of parity errors	Ulong	4	H+32
11	fram err	Number of framing errors	Ulong	4	H+36
12	overruns	Number of hardware overruns	Ulong	4	H+40
13	Next port offset = H + 4 + (#port x 40)				
14	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4+ (#port x 40)
15	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log portstatsa once

ASCII example:

```
#PORTSTATSA,COM1,0,43.0,FINESTEERING,1039,420241.409,00000028,4aa8,0;
3,
COM1,603,1479967,588,0,420160,2,0,15,0,
COM2,1588307,336698,1586999,115099,224314,1308,0,0,0,
COM3,0,0,0,0,6,0,0,0,0*00c49f87
```

3.4.39 PSRDOP Pseudorange DOP

The dilution of precision data is calculated using the geometry of only those satellites that are currently being tracked and used in the position solution by the GPSCard. This log is updated once every 60 seconds or whenever a change in the satellite constellation occurs. Therefore, the total number of data fields output by the log is variable and depends on the number of SVs that are being tracked. Twelve is the maximum number of SV PRNs contained in the list.

✉ 1. If a satellite is locked out using the LOCKOUT command, it will still be shown in the PRN list, but it will be significantly de-weighted in the DOP calculation

2. The vertical dilution of precision can be calculated by: $vdop = \sqrt{pdop^2 - hdop^2}$

Structure: **Message ID: 174**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	gdop	Geometric dilution of precision - assumes 3-D position and receiver clock offset (all 4 parameters) are unknown.	Float	4	H
3	pdop	Position dilution of precision - assumes 3-D position is unknown and receiver clock offset is known.	Float	4	H+4
4	hdop	Horizontal dilution of precision.	Float	4	H+8
5	htdop	Horizontal position and time dilution of precision.	Float	4	H+12
6	tdop	Time dilution of precision - assumes 3-D position is known and only the receiver clock offset is unknown.	Float	4	H+16
7	cutoff	Elevation cut-off angle.	Float	4	H+20
8	#PRN	Number of satellites PRNs to follow.	Long	4	H+24
9	PRN	PRNof SV PRN tracking, null field until position solution available.	Ulong	4	H+28
10...	Next PRN offset = H + 28 + (#prn x 4)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+28+ (#prn x 4)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log psrdopa onchanged

ASCII Example:

```
#PSRDOPA,COM1,0,54.0,COARSE,1037,322184.500,00000028,8244,0;
1.4803,1.3396,0.8670,1.0717,0.6299,0.0,9,1,30,25,18,4,15,16,22,14*8efc1ebe
```

3.4.40 PSRPOS Pseudorange Position

This log contains the pseudorange position computed by the receiver, along with three status flags. In addition, it reports other status indicators, including differential age, which is useful in predicting anomalous behavior brought about by outages in differential corrections.

Structure: **Message ID: 47**

Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status (see *Table 40, Solution Status on Page 124*)	Enum	4	H
3	pos type	Position type (see *Table 39, Position or Velocity Type on Page 123*)	Enum	4	H+4
4	lat	Latitude	Double	8	H+8
5	lon	Longitude	Double	8	H+16
6	hgt	Height above mean sea level	Double	8	H+24
7	undulation	Undulation	Float	4	H+32
8	datum id#	Datum ID number (see *Chapter 2, Table 17, Datum Transformation Parameters on Page 54*)	Enum	4	H+36
9	lat σ	Latitude standard deviation	Float	4	H+40
10	lon σ	Longitude standard deviation	Float	4	H+44
11	hgt σ	Height standard deviation	Float	4	H+48
12	stn id	Base station ID	Char[4]	4	H+52
13	diff_age	Differential age in seconds	Float	4	H+56
14	sol_age	Solution age in seconds	Float	4	H+60
15	#obs	Number of observations tracked	Uchar	1	H+64
16	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+65
17	Reserved		Uchar	1	H+66
18			Uchar	1	H+67
19			Uchar	1	H+68
20			Uchar	1	H+69
21			Uchar	1	H+70
22			Uchar	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log psrposa ontime 1

ASCII Example:

```
#PSRPOSA,COM1,0,58.0,FINE,1027,324231.000,00000000,fc91,0;
SOL_COMPUTED,PSRDIFF,51.11615533807,-114.03850611829,1010.452,-16.271,61,
19.50677306162,14.52973740177,39.584,"0",0.0,60.000,9,8,8,1,0,0,0,0*de152df7
```

3.4.41 PSRVEL Pseudorange Velocity

In the PSRVEL log the actual speed and direction of the GPSCard antenna over ground is provided. The velocity measurements sometimes have a latency associated with them. The time of validity is the time tag in the log minus the latency value. See also the table footnote for velocity logs on *Page 107*.

The velocity status indicates varying degrees of velocity quality. To ensure healthy velocity, the position sol-status must also be checked. If the sol-status is non-zero, the velocity will likely be invalid. It should be noted that the GPSCard does not determine the direction a vessel, craft, or vehicle is pointed (heading), but rather the direction of the motion of the GPS antenna relative to the ground.

The velocity is computed using Doppler values typically derived from differences in consecutive carrier phase measurements. As such, it is an average velocity based on the average change in pseudorange over the time interval and not an instantaneous velocity at the PSRVEL time tag. The velocity latency to be subtracted from the time tag will normally be 1/2 the time between filter updates. Under default operation, the position filter is updated at a rate of 2 Hz. This translates into a velocity latency of 0.25 second. The latency can be reduced by increasing the update rate of the filter by requesting the BESTVEL, PSRVEL, BESTPOS or PSRPOS messages at a rate higher than 2 Hz. For example, a logging rate of 10 Hz would reduce the velocity latency to 0.005 seconds. For integration purposes, the velocity latency should be applied to the record time tag.

A valid solution with a latency of 0.0 indicates that the instantaneous Doppler measurement was used to calculate velocity.

Structure: **Message ID: 100**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	latency	A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results.	Float	4	H+8
5	age	Differential age	Float	4	H+12
6	hor spd	Horizontal speed over ground, in meters per second	Double	8	H+16
7	trk gnd	Actual direction of motion over ground (track over ground) with respect to True North, in degrees	Double	8	H+24
8	vert spd	Vertical speed, in meters per second, where positive values indicate increasing altitude (up) and negative values indicate decreasing altitude (down)	Double	8	H+32
9	Reserved		Float	4	H+40
10	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
11	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log psrvela ontime 1

ASCII Example:

```
#PSRVELA,COM1,0,62.5,FINESTEERING,1049,247680.000,00000128,cf7a,0;
SOL_COMPUTED,DOPPLER_VELOCITY,0.250,1.000,0.112280259,0.0435,54.537543,0.0255
*9f4a42ad
```

3.4.42 PSRXYZ Pseudorange Cartesian Position and Velocity

This log contains the receiver's pseudorange position and velocity in ECEF coordinates. The position and velocity status field's indicate whether or not the corresponding data is valid. See *Figure 6, Page 127* for a definition of the ECEF coordinates.

The velocity measurements sometimes have a latency associated with them. The time of validity is the time tag in the log minus the latency value.

The velocity status indicates varying degrees of velocity quality. To ensure healthy velocity, the position sol-status must also be checked. If the sol-status is non-zero, the velocity will likely be invalid. It should be noted that the receiver does not determine the direction of a vessel, craft, or vehicle is pointed (heading), but rather the direction of the motion of the GPS antenna relative to the ground.

The velocity is computed using Doppler values typically derived from differences in consecutive carrier phase measurements. As such, it is an average velocity based on the average change in pseudorange over the time interval and not an instantaneous velocity at the PSRVEL time tag. The velocity latency to be subtracted from the time tag will normally be 1/2 the time between filter updates. Under default operation, the position filter is updated at a rate of 2 Hz. This translates into a velocity latency of 0.25 second. The latency can be reduced by increasing the update rate of the filter by requesting the BESTXYZ message at a rate higher than 2 Hz. For example, a logging rate of 10 Hz would reduce the velocity latency to 0.005 seconds. For integration purposes, the velocity latency should be applied to the record time tag.

A valid solution with a latency of 0.0 indicates that the instantaneous Doppler measurement was used to calculate velocity.

Structure: **Message ID: 243**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	P-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	P-X	Position X-coordinate (m)	Double	8	H+8
5	P-Y	Position Y-coordinate (m)	Double	8	H+16
6	P-Z	Position Z-coordinate (m)	Double	8	H+24
7	P-X σ	Standard deviation of P-X (m)	Float	4	H+32
8	P- Y σ	Standard deviation of P-Y (m)	Float	4	H+36
9	P-Z σ	Standard deviation of P-Z (m)	Float	4	H+40
10	V-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H+44
11	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+48
12	V-X	Velocity vector along X-axis (m)	Double	8	H+52
13	V-Y	Velocity vector along Y-axis (m)	Double	8	H+60

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
14	V-Z	Velocity vector along Z-axis (m)	Double	8	H+68
15	V-X σ	Standard deviation of V-X (m)	Float	4	H+76
16	V-Y σ	Standard deviation of V-Y (m)	Float	4	H+80
17	V-Z σ	Standard deviation of V-Z (m)	Float	4	H+84
18	stn ID	Base station ID	Char[4]	4	H+88
19	V-latency	A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results.	Float	4	H+92
20	diff_age	Differential age	Float	4	H+96
21	sol_age	Solution age in seconds	Float	4	H+100
22	#obs	Number of observations tracked	Uchar	1	H+104
23	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+105
24	Reserved		Char	1	H+106
25			Char	1	H+107
26			Char	1	H+108
27			Char	1	H+109
28			Char	1	H+110
29			Char	1	H+111
30	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+112
31	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log psrxyza ontime 1

ASCII Example:

```
#PSRXYZA,COM1,0,68.5,FINESTEERING,1058,405020.000,00000028,ab38,37;
SOL_COMPUTED,SINGLE,-1634528.841,-3664567.840,4942460.135,25.3018,26.8277,
37.8962,SOL_COMPUTED,DOPPLER_VELOCITY,-0.049,0.110,0.344,3.0241,3.2065,
4.5295,"",0.250,0.000,0.000,7,7,0,0,0,0,0,0*2411d87d
```

3.4.43 RANGE Satellite Range Information

RANGE contains the channel measurements for the currently tracked satellites. When using this log, please keep in mind the constraints noted along with the description.

It is important to ensure that the receiver clock has been set. This can be monitored by the bits in the *Receiver Status* field of the log header. Large jumps in pseudorange as well as accumulated Doppler range (ADR) will occur as the clock is being adjusted. If the ADR measurement is being used in precise phase processing, it is important not to use the ADR if the "parity known" flag in the *ch-tr-status* field is not set as there may exist a half (1/2) cycle ambiguity on the measurement. The tracking error estimate of the pseudorange and carrier phase (ADR) is the thermal noise of the receiver tracking loops only. It does not account for possible multipath errors or atmospheric delays.

If both the L1 and L2 signals are being tracked for a given PRN, two entries with the same PRN will appear in the range logs. As shown in *Table 53*, these entries can be differentiated by bit 20, which is set if there are multiple observables for a given PRN, and bits 21-22, which denotes whether the observation is for L1 or L2. This is to aid in parsing the data.

Table 53: Channel Tracking Status

Nibble #	Bit #	Mask	Description	Range Value
N0	0	0x00000001	Tracking state	0-11, see *Table 55, Tracking State on Page 179*
	1	0x00000002		
	2	0x00000004		
	3	0x00000008		
N1	4	0x00000010		
	5	0x00000020	SV channel number	0-n (0 = first, n = last) n depends on GPSCard
	6	0x00000040		
	7	0x00000080		
N2	8	0x00000100		
	9	0x00000200		
	10	0x00000400	Phase lock flag	0 = Not locked, 1 = Locked
	11	0x00000800	Parity known flag	0 = Not known, 1 = Known
N3	12	0x00001000	Code locked flag	0 = Not locked, 1 = Locked
	13	0x00002000	Correlator spacing	0-7, see *Table 56, Correlator Spacing on Page 179*
	14	0x00004000		
	15	0x00008000		
N4	16	0x00010000	Satellite system	0 = GPS, 1, 3-7 = Reserved, 2 = WAAS
	17	0x00020000		
	18	0x00040000		
	19	0x00080000	Reserved	
N5	20	0x00100000	Grouping [a]	0 = Not grouped, 1 = Grouped
	21	0x00200000	Frequency	0 = L1, 1 = L2, 2-3 = Reserved
	22	0x00400000		
	23	0x00800000	Code type	0 = C/A, 1 = P, 2 = P codeless 3-7 = Reserved
N6	24	0x01000000		
	25	0x02000000		
	26	0x04000000	Forward Error Correction (FEC)	0 = Not FEC, 1 = FEC
	27	0x08000000	Primary L1 channel	0 = Not primary, 1 = Primary
N7	28-30		Reserved	
	31	0x80000000	Channel assignment	0 = Automatic, 1 = Forced

a. Grouped: Channel has an associated channel (L1/L2 pairs)

Structure: **Message ID: 43**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	# obs	Number of observations with information to follow.[a]	Long	4	H
3	PRN	GPS satellite PRN number of range measurement.	UShort	2	H+4
4	Reserved		UShort	2	H+6
5	psr	Pseudorange measurement (m).	Double	8	H+8
6	psr std	Pseudorange measurement standard deviation (m).	Float	4	H+16
7	adr	Carrier phase, in cycles (accumulated Doppler range).	Double	8	H+20
8	adr std	Estimated carrier phase standard deviation (cycles).	Float	4	H+28
9	dopp	Instantaneous carrier Doppler frequency (Hz).	Float	4	H+32
10	C/N_0	Carrier to noise density ratio $C/N_0 = 10[\log_{10}(S/N_0)]$ (dB-Hz)	Float	4	H+36
11	locktime	Number of seconds of continuous tracking (no cycle slipping).	Float	4	H+40
12	ch-tr-status	Tracking status (see *Channel Tracking Status on Page 177* and the example in *Table 54*).	ULong	4	H+44
13...	Next PRN offset = H + 4 + (#obs x 44)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4+ (#obs x 44)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. Some satellite PRN numbers may have two lines of observations, one for the L1 frequency and the other for L2.

Recommended Input:

log rangea ontime 30

ASCII Example:

```
#RANGEA,COM1,0,54.5,FINESTEERING,1039,422126.500,00000028,5b72,0;
17,
18,0,22308014.258,0.006,-122201546.725,0.000,-3424.688,47.8,9795.172,8105c24,
...
10,0,23513127.359,0.026,-96282262.554,0.000,-600.467,32.0,7007.400,1305d6b
*bc7d38a8
```

Table 54: Channel Tracking Example

	N7				N6				N5			N4				N3				N2		N1				N0			
0x	0				8				1			0				5				C		2				4			
Bit #	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Binary [a]	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0
Data	Chan. Assign-ment	Reserved (R)			Primary L1	R	Code Type			Frequency		Grouping	R	Satellite System			Correlator Spacing			Code locked flag	Parity flag	Phase lock flag	Channel Number					Tracking State			
Value	Auto-matic				Primary		C/A			L1		Grouped		GPS			Narrow < 1 chip			Locked	Known	Locked	Channel 1					L1 Phase Lock Loop			

a. For a complete list of hexadecimal and binary equivalents please refer to the appendix on *Unit Conversion* in *Volume 1* of this manual.

Table 55: Tracking State

State	Description	State	Description
0	L1 Idle	6	L1 Steering
1	L1 Sky search	7	L1 Frequency-lock loop
2	L1 Wide frequency band pull-in	8	L2 Idle
3	L1 Narrow frequency band pull-in	9	L2 P-code alignment
4	L1 Phase lock loop	10	L2 Search
5	L1 Reacquisition	11	L2 Phase lock loop

Table 56: Correlator Spacing

State	Description
0	Reserved
1	Standard correlator: spacing = 1 chip
2	Narrow Correlator: spacing < 1 chip
3	Reserved
4	Pulse Aperture Correlator (PAC)

3.4.44 RANGECMP Compressed Version of the RANGE Log

Structure: **Message ID: 140**
Log Type: Synch

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#obs	Number of satellite observations with information to follow.	Long	4	H
3	1st range record	Compressed range log in format of *Table 57*	Hex	24	H+4
4	Next rangecmp offset = H + 4 + (#obs x 24)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H + 4 + (#obs x 24)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rangecmpa ontime 10

Example:

```
#RANGECMPA,COM1,0,49.0,FINESTEERING,1039,231811.800,00000028,21a2,0;
20,
08105c04006df200a106c072fff83a1400010b2527800000,
01305c0b0055abf0a106c15effbfb6dd00010b0c36000000,
08105c2400d77c80b0ad3878fffbbb7a00160aec25c00000,
01305c2b00a7e960b0ad3adbff93772f00160ab734000000,
08105c440093f680bbc380ccff9722a5000405e955800000,
01305c4b00734bb0bbc38125ffc741901004050db2400000,
0810546400d9d600c20edd0dffefa1473412002db4000000,
0130546b00a9bdf0c20edd3cffbaa041a9120012b2800000,
08105ca4ff0be080bbea553dffdd83a7000f0b1a6d400000,
01305cabff41c620bbea589bff89b3f5000f0b05ab000000,
08105ce4001cea809b36d1c6ff8e8404000e0b2024c00000,
...,
01305d6bffa39210b6bf5a79ffdc683c001e0b0732c00000*1ef96252
```

Table 57: Range Record Format (RANGECMP only)

Data	Bit(s) first to last	Length (bits)	Scale Factor	Units
Channel Tracking Status	0-31	32	see *Table 53, Channel Tracking Status on Page 177*	-
Doppler Frequency	32-59	28	1/256	Hz
Pseudorange (PSR)	60-95	36	1/128	m
ADR[a]	96-127	32	1/256	cycles
StdDev-PSR	128-131	4	see[b]	m
StdDev-ADR	132-135	4	(n + 1)/512	cycles
PRN	136-143	8	1	-
Lock Time[c]	144-164	21	1/32	s
C/No[d]	165-169	5	(20 + n)	dB-Hz
Reserved	170-191	22		

a. ADR (Accumulated Doppler Range) is calculated as follows:

ADR_ROLLS = (RANGECMP_PSR / WAVELENGTH + RANGECMP_ADR) / MAX_VALUE

Round to the closest integer

IF (ADR_ROLLS ≤ 0)
    ADR_ROLLS = ADR_ROLLS - 0.5
ELSE
    ADR_ROLLS = ADR_ROLLS + 0.5

At this point integerise ADR_ROLLS

CORRECTED_ADR = RANGECMP_ADR - (MAX_VALUE*ADR_ROLLS)

where

ADR has units of cycles
WAVELENGTH = 0.1902936727984 for L1
WAVELENGTH = 0.2442102134246 for L2
MAX_VALUE = 8388608

b.

Code	StdDev-PSR (m)
0	0.050
1	0.075
2	0.113
3	0.169
4	0.253
5	0.380
6	0.570
7	0.854
8	1.281
9	2.375
10	4.750
11	9.500
12	19.000
13	38.000
14	76.000
15	152.000

c. Lock time rolls over after 2,097,151 seconds.

d. C/No is constrained to a value between 20-51 dB-Hz. Thus, if it is reported that C/No = 20 dB-Hz, the actual value could be less. Likewise, if it is reported that C/No = 51 dB-Hz, the true value could be greater.

3.4.45 RAWALM Raw Almanac Data

This log contains the undecoded almanac subframes as received from the satellite. For more information on Almanac data, refer to the GPS SPS Signal Specification (see the appendix on *Standards and References* in *Volume 1* of this manual).

The OEM4 family of receivers will automatically save almanacs in their non-volatile memory (NVM), therefore creating an almanac boot file would not be necessary.

Structure: **Message ID: 74**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	ref week	Almanac reference week number.	Ulong	4	H
3	ref secs	Almanac reference time (seconds.)	Ulong	4	H+4
4	subframes	Number of subframes to follow	Ulong	4	H+8
5	prn	Satellite PRN number	UShort	2	H+12
6	data	Subframe page data.	Hex	30	H+14
7...	Next subframe offset = H + 12 + (subframe x 32)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H + 12 + (32 x subframes)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rawalma onchanged

ASCII Example:

```
#RAWALMA,COM1,0,80.5,SATTIME,1156,411744.000,00000000,076b,614;
1156,589824.000,42,
1,8b021085ccb6412bf2901073fd4400a10d01cae5cdbabbfc5f960a1c003c,
25,8b021085ceb2594f3490fdd0fd4f00a10c9df1e5a3b369ea7d0e52030007,
2,8b021085cf3742b6f790f9affd3500a0ffa11bfe77afd32a2922bceaffd2,
26,8b021085d1325a6e04901248fd4600a10c58ca3f290d1263a4af1ff4ff87,
...
23,8b02108603b4578294901a69fd4d00a10b37a18000b8528b0fd1e3010003,
24,8b0210860637584bbb901949fd6c00a10d18767dc3c032de5d856d01003e,
63,8b02108608327f99999999990990990990999999999908000000000fe9,
51,8b02108608b473908400000000000000003f00003f000fc0000000aaaaaa*10591407
```

3.4.46 RAWEPHEM Raw Ephemeris

This log contains the raw binary information for subframes one, two and three from the satellite with the parity information removed. Each subframe is 240 bits long (10 words - 24 bits each) and the log contains a total 720 bits (90 bytes) of information (240 bits x 3 subframes). This information is preceded by the PRN number of the satellite from which it originated. This message will not be generated unless all 10 words from all 3 frames have passed parity.

Ephemeris data whose TOE (Time Of Ephemeris) is older than six hours will not be shown.

Structure: **Message ID: 41**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Satellite PRN number	Ulong	4	H
3	ref week	Ephemeris reference week number.	Ulong	4	H+4
4	ref secs	Ephemeris reference time (seconds.)	Ulong	4	H+8
5	subframe1	Subframe 1 data.	Hex	30	H+12
6	subframe2	Subframe 2 data.	Hex	30	H+42
7	subframe3	Subframe 3 data.	Hex	30	H+72
8	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+102
9	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rawephema onchanged

ASCII Example:

```
#RAWEPHEMA,COM1,0,70.0,FINESTEERING,1023,512190.000,00000000,2e9f,0;
7,1023,518400,
8b0ffca6bb24ffd70173931172f3b03c9c99c678fc267e9000ffe14b88a0,
8b0ffca6b92b26087a3187cbf13a9e079605828cd00ee8a10d73527e907e,
8b0ffca6b9ad0088f0526ba5005a26e038131e06a8925200ffa76826fcd8*2ddcf8b5
```

3.4.47 RAWGPSSUBFRAME Raw Subframe Data

This log contains the raw GPS subframe data.

Structure: **Message ID: 25**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	decode #	Frame decoder number	Ulong	4	H
3	PRN	Satellite PRN number	Ulong	4	H+4
4	subfr id	Subframe ID	Ulong	4	H+8
5	data	Raw subframe data	Hex[30]	32[a]	H+12
6	chan	Signal channel number that the frame was decoded on.	Ulong	4	H+44
7	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+48
8	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rawgpssubframea onnew

ASCII Example:

```
#RAWGPSSUBFRAME,COM1,29,58.5,SATTIME,1047,228642.000,00000128,d5dc,0;
0,1,3,8b005c4a6e2e0033f78c7961002a271db3121674b8fc3651ffab58781417,
0*b0fc82c4
...
#RAWGPSSUBFRAME,COM1,11,58.5,SATTIME,1047,228654.000,00000128,d5dc,0;
18,24,5,8b005c4a6f36566a4d63fa5dfd2f00a10dde49d00c1628f842e7a0320029,
18*6f03db2a
```

3.4.48 RAWGPSWORD Raw Navigation Word

This message contains the framed raw navigation words. Each log contains a new 30 bit navigation word (in the least significant 30 bits), plus the last 2 bits of the previous word (in the most significant 2 bits). The 30 bit navigation word contains 24 bits of data plus 6 bits of parity. The GPS time stamp in the log header is the time that the first bit of the 30 bit navigation word was received. Only navigation data that has passed parity checking will appear in this log. One log will appear for each PRN being tracked every 0.6 seconds if logged ONNEW or ONCHANGED.

Structure: **Message ID: 407**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	PRN	Satellite PRN number	Ulong	4	H
3	nav word	Raw navigation word	Ulong	4	H+4
4	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+8
5	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rawgpsworda onnew

ASCII Example:

```
#RAWGPSWORDA,COM1,0,76.5,FINESTEERING,1117,334881.673,00000000,3eef,33158;
26,64460e5b*25e8c10b
```

3.4.49 RAWOMNIFRAME Raw OmniSTAR Frame Data

This log contains the raw OmniSTAR frame data.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Structure: **Message ID: 441**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	data	Raw OmniSTAR frame data.	Uchar[1200]	1200	H
3	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+1200
4	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rawomniframea onnew

ASCII Example:

```
#RAWOMNIFRAMEA,COM1,0,63.5,FINESTEERING,1160,406775.132,00000000,e96c,690;
474,41,79,46,47,37,8e,25,25,69,30,2b,7d,97,94,9a,61,9c,9d,50,5a,6f,9e,8f,62,3
1,73,30,81,88,84,60,7c,7e,81,69,26,90,5e,6c,97,7f,7b,88,86,26,51,35,8d,52,6e,
5f,3a,8e,8d,38,9f,5b,38,20,20,9b,9d,6c,2e,9b,23,97,6e,85,60,30,7e,23,8d,57,84
,5e,95,6e,7f,72,52,4e,81,6f,30,98,49,6f,50,43,6c,6c,5a,22,63,85,5d,74,9f,34,9
7,39,73,86,47,87,9c,40,80,7c,0d,3f,a0,7d,20,91,24,68,42,64,6a,20,44,9f,9f,85,
67,9d,82,20,20,20,20,25,20,5f,8e,23,9f,30,3f,8f,21,9f,80,2f,9f,20,20,74,20,21
,20,5f,7c,20,21,90,20,2b,20,21,28,22,9b,2f,8d,83,9d,4b,20,22,62,20,3d,30,3f,7
6,21,99,80,20,6e,60,26,78,20,4c,20,21,68,20,38,60,27,9f,9f,8e,67,9c,6a,20,9f,
40,20,26,60,5f,78,20,21,70,20,31,20,22,60,20,3b,20,21,7c,27,9f,20,5f,56,20,23
,80,20,47,20,22,28,24,24,20,39,63,9a,5d,20,33,9f,9e,90,2f,98,94,20,25,28,20,5
6,20,20,28,27,92,60,5d,96,23,94,30,3f,34,20,23,30,20,51,60,9d,74,20,32,40,20,
44,20,2a,60,2a,28,20,8c,67,95,32,20,58,20,28,24,3f,8a,78,22,6e,5f,81,58,9f,3e
,20,20,51,60,9b,7c,20,36,40,21,86,20,37,60,21,97,20,2e,28,20,59,60,25,54,20,4
7,20,51,9f,9f,74,2f,9a,80,20,21,68,20,2d,60,20,34,27,9b,20,5f,3c,23,98,80,3f,
4c,21,9f,68,20,2c,60,9f,94,20,25,80,5f,80,20,21,70,3c,9b,9d,29,27,9b,8a,20,9b
,8c,27,9e,40,20,42,20,25,70,20,42,20,22,48,2f,94,20,21,2c,27,93,20,5f,8e,23,9
2,80,3e,60,22,2f,27,9c,8a,20,9b,3c,20,64,4f,0d,3f,68,41,20,3b,95,21,44,20,48,
25,2e,31,22,2a,83,37,57,8e,7c,20,41,5e,70,44,9e,2f,5f,9c,9b,27,3f,2e,81,80,8f
,7f,22,42,96,21,53,29,70,33,71,5f,69,32*794ce370
```

3.4.50 RAWOMNIPACKET Raw OmniSTAR Data Packet

This log contains the raw OmniSTAR packet data.

✉ In addition to a NovAtel OEM4 card, a NovAtel OmniSTAR L-Band receiver and a subscription to the OmniSTAR service is required. Contact NovAtel for details. Contact information may be found on the back of this manual or you can refer to the *Customer Service* section in *Volume 1* of this manual.

Structure: **Message ID: 457**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	data	Raw OmniSTAR data packet.	Uchar[128]	128	H
3	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+128
4	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rawomnipacketa onnew

ASCII Example:

```
#RAWOMNIPACKETA,COM1,0,62.0,FINESTEERING,1160,406879.140,00000000,b28f,690;
20,3b,7e,9e,77,49,3f,77,75,47,4a,9f,95,98,32,3d,52,2c,3e,6c,70*9088aad6
```

3.4.51 RAWWAASFRAME Raw SBAS Frame Data SBAS

This log contains the raw SBAS frame data of 226 bits (8-bit preamble, 6-bit message type and 212 bits of data but without a 24-bit CRC). Only frame data with a valid preamble and CRC will be reported.

Structure: **Message ID: 287**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	decode #	Frame decoder number	Ulong	4	H
3	PRN	SBAS satellite PRN number	Ulong	4	H+4
4	WAASmsg id	SBAS frame ID	Ulong	4	H+8
5	data	Raw SBAS frame data. There are 226 bits of data and 6 bits of padding.	Uchar[29]	32[a]	H+12
6	chan	Signal channel number that the frame was decoded on.	Ulong	4	H+44
7	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+48
8	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rawwaasframea onnew

ASCII Example:

```
#RAWWAASFRAMEA,COM1,13,62.5,SATTIME,1093,494299.000,00000020,b2c6,209;11,122,
24,9a63ff9ffdffc00bfe8010fb96efa0873dffbebfd401a020003fef37c0,11*7a178380
#RAWWAASFRAMEA,COM1,12,59.5,SATTIME,1093,494300.000,00000020,b2c6,209;11,122,
0,c600fff5fffff8001ffdffdffc015fffff1ffdffc00078cfbb979bb900,11*3fa9148a
#RAWWAASFRAMEA,COM1,11,59.5,SATTIME,1093,494301.000,00000020,b2c6,209;11,122,
3,530cdffdffdffdfffffec001fffffdffc010009ffffe7bbb8cb8f8878c0,11*9a8f7575
#RAWWAASFRAMEA,COM1,10,59.5,SATTIME,1093,494302.000,00000020,b2c6,209;11,122,
26,9a68c0d9e5ff2678dfc6de48f1e78d7c6de000000000000000000000000,11*68e08451
...
#RAWWAASFRAMEA,COM1,0,62.0,SATTIME,1093,494316.000,00000020,b2c6,209;11,122,6
3,53fc0000000000000000000000000000000000000000000000000000000000,11*a9fb31c3
#RAWWAASFRAMEA,COM1,0,62.5,SATTIME,1093,494317.000,00000020,b2c6,209;11,122,2
4,9a63ff5ffdffc003fe8020fb92efa0a43e0680100c036000201fef38c0,11*c1934588
#RAWWAASFRAMEA,COM1,0,62.5,SATTIME,1093,494318.000,00000020,b2c6,209;11,122,0
,c600fff1fffffc005ffdffdffc015fffff1ffdffc00478cfbb979bb900,11*261e7fb0
```

3.4.52 REFSTATION Base Station Position and Health RTK

This log contains the ECEF Cartesian position of the base station as received through the RTCM, RTCA, or CMR message. It also features a time tag, the health status of the base station, and the station ID. This information is set at the base station using the FIX POSITION command and the CMR/RTCA/RTCMTXSTATION command.

Structure: **Message ID: 175**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	status	Status of the base station information (see *Table 58* below).	ULong	4	H
3	x	ECEF X value.	Double	8	H+4
4	y	ECEF Y value.	Double	8	H+12
5	z	ECEF Z value.	Double	8	H+20
6	health	Base station health.	Ulong	4	H+28
7	stn type	Base station type (see *Table 59 on Page 189*).	Enum	4	H+32
8	stn ID	Base station ID.	Char[5]	8[a]	H+36
9	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
10	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log refstationa onchanged

ASCII Example:

```
#REFSTATIONA,COM1,0,60.5,FINESTEERING,1037,507040.202,00000028,6c48,0;
00000000,-1634526.770,-3664610.130,4942483.610,6,RTCM,"10"*43e693fe
```

Table 58: Base Station Status

Bit #	Mask	Description	Bit = 0	Bit = 1
0	0x00000001	Validity of the base station.	Valid	Invalid

Table 59: Base Station Type

Base Station Type (Binary)	Base Station Type (ASCII)	Description
0	NONE	Base station is not used
1	RTCM	Base station is RTCM
2	RTCA	Base station is RTCA
3	CMR	Base station is CMR

3.4.53 RTCA Standard Logs *DGPS*

RTCA1 DIFFERENTIAL GPS CORRECTIONS

Message ID: 10

RTCAEPHEM EPHEMERIS AND TIME INFORMATION

Message ID: 347

RTCAOBS BASE STATION OBSERVATIONS *RTK*

Message ID: 6

RTCAREF BASE STATION PARAMETERS *RTK*

Message ID: 11

The RTCA (Radio Technical Commission for Aviation Services) Standard is being designed to support Differential Global Navigation Satellite System (DGNSS) Special Category I (SCAT-I) precision instrument approaches. The RTCA Standard is in a preliminary state. NovAtel's current support for this Standard is based on "Minimum Aviation System Performance Standards DGNSS Instrument Approach System: SCAT-I" dated August 27, 1993.

✉ The above messages can be logged with an A or B suffix for an ASCII or Binary output with a NovAtel header followed by Hex or Binary raw data respectively.

See the chapter on *Message Formats* in *Volume 1* of this manual for more information on RTCA standard logs.

Example Input:

```
interfacemode com2 none RTCA
fix position 51.1136 -114.0435 1059.4
log com2 rtcaobs ontime 2
log com2 rtcaref ontime 10
log com2 rtca1 ontime 10 3
log com2 rtcaephem ontime 10 7
```

3.4.54 RTCADATA1 Differential GPS Corrections DGPS

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCA standard logs.

Structure: **Message ID: 392**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	z-count	Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris.	Double	8	H
3	AEB	Acceleration Error Bound	Uchar	4 [a]	H+8
4	#prn	Number of satellite corrections with information to follow	Ulong	4	H+12
5	prn	PRN number of satellite	Ulong	4	H+16
6	range	Pseudorange correction (m)	Double	8	H+20
7	IODE	Issue of ephemeris data	Uchar	4 [a]	H+28
8	range rate	Pseudorange rate correction (m/s)	Double	8	H+32
9	UDRE	User differential range error	Float	4	H+40
10...	Next prn offset = H+16 + (#prns x 28)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rtcadata1a ontime 10 3

ASCII Example:

```
#RTCADATA1A,COM1,0,73.5,FINESTEERING,1117,159373.000,00100020,4794,399;
973.000000000,0,11,
3,-11.635082719,43,-0.005679527,1.000000000,
15,-4.791180559,96,-0.008062044,1.000000000,
18,-2.955923688,1,-0.006305533,1.000000000,
21,-21.557364315,153,-0.012775200,1.000000000,
17,-0.764539172,18,-0.018026272,1.000000000,
6,-20.653744978,88,-0.011639515,1.000000000,
26,-6.605308463,35,-0.007333527,1.000000000,
23,-1.911155288,167,-0.006073328,1.000000000,
28,-31.304769386,22,-0.001623565,1.000000000,
10,-36.312368029,142,-0.021844717,1.000000000,
22,-23.069976260,27,-0.007292280,1.000000000*65fade5c
```

3.4.55 RTCADATAEPHEM Ephemeris and Time Information DGPS & RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCA standard logs.

Structure: **Message ID: 393**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	des	NovAtel designator.	Uchar	1	H
3	subtype	RTCA message subtype	Uchar	3[a]	H+1
4	week	GPS week number (weeks)	Ulong	4	H+4
5	sec	Seconds into the week (seconds)	Ulong	4	H+8
6	prn	PRN number	Ulong	4	H+12
7	Reserved		Uchar	4[b]	H+16
8	raw data	Raw ephemeris data	Hex[90]	92[a]	H+20
9	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+112
10	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment
b. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rtcadataephema ontime 10 7

ASCII Example:

```
#RTCADATAEPHEMA,COM1,0,73.5,FINESTEERING,1117,159879.363,00100020,1cab,399;
78,2,12,159879,28,0,8b0174340b2717520225c6b86ef0c9b885c3c315ea16286e00ffed006
98b8b017434092816015d2f38c3da966d010e029f09a2161ca1172de3286e7c8b01743409ae00
63e69be1f7ffd7271ba3bd16679e0ae8f1ffab9816e5c4*eeac48cc
```

3.4.56 RTCADATAOBS Base Station Observations RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCA standard logs.

Structure: **Message ID: 394**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	des	NovAtel designator.	Uchar	1	H
3	subtype	RTCA message subtype	Uchar	3[a]	H+1
4	min psr	Minimum pseudorange	Double	8	H+4
5	sec	Seconds into the GPS week (seconds)	Float	4	H+12
6	Reserved		Long	4	H+16
7	#prn	Number of PRNs with information to follow.	Ulong	4	H+20
8	trans ID	Transmitter ID	Uchar	1	H+24
9	L1 lock	L1 lock flag	Uchar	1	H+25
10	L2 lock	L2 lock flag	Uchar	2[b]	H+26
11	L1 psr	L1 pseudorange offset (2/10 m)	Double	8	H+28
12	L2 psr	L2 pseudorange offset (1/4 m)	Double	8	H+36
13	L1 ADR	L1 carrier phase offset, accumulated Doppler range (2/1000 m)	Float	4	H+44
14	L2 ADR	L2 carrier phase offset, accumulated Doppler range (3/1000 m)	Float	4	H+48
15	L2 encrypt	L2 not encrypted 0 = FALSE 1 = TRUE?	Enum	4	H+52
16	Reserved		Long	4	H+56
17...	Next PRN offset = H+24 + (#prns x 36)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment
b. In the binary log case an additional 1 byte of padding is added to maintain 4 byte alignment

Recommended Input:

log rtcadataobsa ontime 2

ASCII Example:

```
#RTCADATAOBSA,COM1,0,72.0,FINESTEERING,1117,159937.000,00100020,9c6d,399;
78,1,2.010422200000000e+07,337.000000000,0,10,
17,3,3,73257.000000000,2.750000000,0.284000009,0.507000029,TRUE,16,
23,3,3,579946.200000000,3.000000000,-0.606000006,-1.641000032,TRUE,16,
15,3,3,0.000000000,2.750000000,-0.101999998,-0.828000009,TRUE,16,
18,3,3,658823.800000000,1.750000000,-0.425999999,-1.781999946,TRUE,16,
26,3,3,1336423.400000000,3.500000000,0.284000009,0.465000004,TRUE,16,3,3,3,26
93282.400000000,4.000000000,-1.764000058,-6.228000164,TRUE,16,
21,3,3,3574215.600000000,5.750000000,-0.544000030,-4.394999981,TRUE,16,
6,3,3,3216619.400000000,6.000000000,2.592000008,8.819999695,TRUE,16,
22,3,3,4119377.800000000,6.500000000,2.654000044,9.656999588,TRUE,16,28,3,3,4
987382.400000000,6.000000000,-0.032000002,-1.052999973,TRUE,16*1290a33c
```

3.4.57 RTCADATAREF Base Station Parameters RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCA standard logs.

Structure: **Message ID: 395**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	des	NovAtel designator.	Uchar	1	H
3	subtype	RTCA message subtype	Uchar	3[a]	H+1
4	X pos	Base station X coordinate position (mm)	Double	8	H+4
5	Y pos	Base station Y coordinate position (mm)	Double	8	H+12
6	Z pos	Base station Z coordinate position (mm)	Double	8	H+20
7	Reserved		Uchar	4[b]	H+28
8	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+32
9	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment
b. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rtcadatarefa ontime 10

ASCII Example:

```
#RTCADATAREFA,COM1,0,74.5,FINESTEERING,1117,160107.740,00100020,564c,399;
78,0,-1634525.357607752,-3664610.762499783,4942483.614689490,0*14d7f4ec
```

3.4.58 RTCM Standard Logs *DGPS*

RTCM1 DIFFERENTIAL GPS CORRECTIONS
Message ID: 107

RTCM3 BASE STATION PARAMETERS *RTK*
Message ID: 117

RTCM9 PARTIAL DIFFERENTIAL GPS CORRECTIONS, Euro4 with external oscillator only
Message ID: 275

RTCM15 IONOSPHERIC CORRECTIONS
Message ID: 307

RTCM16 SPECIAL MESSAGE
Message ID: 129

RTCM16T SPECIAL TEXT MESSAGE, *see also Page 96*
Message ID: 131

RTCM1819 RAW MEASUREMENTS *RTK*
Message ID: 260

RTCM2021 MEASUREMENT CORRECTIONS *RTK*
Message ID: 374

RTCM22 EXTENDED BASE STATION *RTK*
Message ID: 118

RTCM59 TYPE 59N-0 NOVATEL PROPRIETARY RT20 DIFFERENTIAL *RTK*
Message ID: 116

The Radio Technical Commission for Marine Services (RTCM) was established to facilitate the establishment of various radio navigation standards, which includes recommended GPS differential standard formats.

The Standard recommended by the RTCM Special Committee 104, Differential GPS Service (RTCM SC-104, Washington, D.C.) have been adopted by NovAtel for implementation into the GPSCard. The GPSCard can easily be integrated into positioning systems around the globe because it is capable of utilizing RTCM formats.

✉ The above messages can be logged with an A or B suffix for an ASCII or Binary output with a NovAtel header followed by Hex or Binary raw data respectively.

See the chapter on *Message Formats* in *Volume 1* of this manual for more information on RTCM standard logs.

Example Input:

```
interfacemode com2 none RTCM
fix position 51.1136 -114.0435 1059.4
log com2 rtcm3 ontime 10
log com2 rtcm22 ontime 10
log com2 rtcm1819 ontime 2
```

3.4.59 RTCMDATA1 Differential GPS Corrections DGPS

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 396**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	#prn	Number of PRNs with information to follow	Ulong	4	H+24
9	scale	Scale where 0 = 0.02 m and 0.002 m/s 1 = 0.32 m and 0.032 m/s	Ulong	4	H+28
10	UDRE	User differential range error	Ulong	4	H+32
12	prn	Satellite PRN number	Ulong	4	H+36
12	psr corr	Scaled pseudorange correction (meters)	Long	4	H+40
13	rate corr	Scaled range rate correction	Long	4	H+44
14	IOD	Issue of data	Long	4	H+48
15...	Next PRN offset = H+28 + (#prns x 24)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata1a ontime 10 3

ASCII Example:

```
#RTCMDATA1A,COM1,0,73.0,FINESTEERING,1117,160453.000,00100020,5745,399;
1117,0,3421,8461020,1730644,6,
9,
0,0,3,-545,-46,43,
0,0,15,-313,-44,96,
0,0,18,-112,-41,1,
0,0,21,-874,-43,153,
0,0,6,-1368,-43,88,
0,0,26,-398,-43,35,
0,0,23,-123,-43,167,
0,0,28,-1302,-39,22,
0,0,22,-1515,-48,27*b60bf22f
```

3.4.60 RTCMDATA3 Base Station Parameters *RTK*

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 402**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	ECEF-X	Base station ECEF X-coordinate (1/100 m)	Double	8	H+24
9	ECEF-Y	Base station ECEF Y-coordinate (1/100 m)	Double	8	H+32
10	ECEF-Z	Base station ECEF Z-coordinate (1/100 m)	Double	8	H+40
11	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+48
12	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata3a ontime 10

ASCII Example:

```
#RTCMDATA3A,COM1,0,74.0,FINESTEERING,1117,160636.477,00100020,2e19,399;
1117,0,3727,160636477,180,6,
-163452535.7607752382755280,-366461076.2499782443046570,
494248361.4689489603042603*f621f163
```

3.4.61 RTCMDATA9 Partial Differential GPS Corrections DGPS

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs. This log is the same as the RTCMDATA1 log but there will only be corrections for a maximum of 3 satellites.

Structure: **Message ID: 404**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	#prn	Number of PRNs with information to follow (maximum of 3)	Ulong	4	H+24
9	scale	Scale where 0 = 0.02 m and 0.002 m/s 1 = 0.32 m and 0.032 m/s	Ulong	4	H+28
10	UDRE	User differential range error	Ulong	4	H+32
11	prn	Satellite PRN number	Ulong	4	H+36
12	psr corr	Scaled pseudorange correction (meters)	Long	4	H+40
13	rate corr	Scaled range rate correction	Long	4	H+44
14	IOD	Issue of data	Long	4	H+48
15...	Next PRN offset = H+28 + (#prns x 24)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata9a ontime 10

ASCII Example:

```
#RTCMDATA9A,COM1,0,74.0,FINESTEERING,1117,160710.000,00100020,8265,399;
160710000,0,3850,0,1117,6,
3,
0,0,21,-866,-29,153,
0,0,6,-1438,-29,88,
0,0,26,-409,-30,35*818597db
```

3.4.62 RTCMDATA15 Ionospheric Corrections DGPS

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 397**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	#prn	Number of PRNs with information to follow	Ulong	4	H+24
9	Reserved		Ulong	4	H+28
10	sat type	Satellite type where 0 = GPS 1 = GLONASS	Ulong	4	H+32
11	prn	Satellite PRN number	Ulong	4	H+36
12	ion delay	Ionospheric delay (cm)	Ulong	4	H+40
13	ion rate	Ionospheric rate (0.05 cm / min.)	Long	4	H+44
14...	Next PRN offset = H+28 + (#prns x 20)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata15a ontime 10

ASCII Example:

```
#RTCMDATA15A,COM1,0,74.5,FINESTEERING,1117,160783.000,00100020,9601,399;
15,0,3971,7799968,5163500,6,
10,
0,0,3,1631,445,
0,0,15,1423,-222,
0,0,18,1275,-334,
0,0,21,1763,-334,
0,0,17,1454,-556,
0,0,6,2063,0,
0,0,26,1579,222,
0,0,23,1423,-111,
0,0,28,1874,445,
0,0,22,2146,-445*19ed193f
```

3.4.63 RTCMDATA16 Special Message DGPS & RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 398**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	#chars	Number of characters to follow	Ulong	4	H+24
9	character	Character	Char	4[a]	H+28
10...	Next PRN offset = H+28 + (#chars x 4)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rtcmdata16a once

ASCII Example:

```
#RTCMDATA16A,COM1,0,66.0,FINESTEERING,1117,161024.000,00100020,e639,399;
161024000,0,4373,2243048,6958196,6,21,“Base station will shut down in 1 hour”
*b6202f15
```

3.4.64 RTCMDATA1819 Raw Measurements RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 399**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header (for RTCM18)	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	freq	Frequency indicator where 0 = L1 2 = L2 (1 is reserved for future use)	Ulong	4	H+24
9	Reserved		Ulong	4	H+28
10	GNSS time	Global Navigation Satellite System (GNSS) time of measurement (microseconds)	Long	4	H+32
11	#obs	Number of observation with information to follow	Long	4	H+36
12	multi bit	Multiple message indicator	Ulong	4	H+40
13	code	Is code P Code? 0 = FALSE 1 = TRUE	Ulong	4	H+44
14	sat type	Satellite type 0 = GPS 1 = GLONASS	Ulong	4	H+48
15	prn	Satellite PRN number	Ulong	4	H+52
16	quality	Data quality indicator, see *Table 60, RTCM1819 Data Quality Indicator on Page 204*	Ulong	4	H+56
17	continuity	Cumulative loss of continuity indicator with a loss of lock counter	Ulong	4	H+60
18	phase	Carrier phase (1/256 cycles)	Long	4	H+64
19...	Next RTCM18 observation offset = H+40 + (#obs x 28)				

variable	RTCM header (for RTCM19)	RTCM message type	Ulong	4	variable
		Base station ID	Ulong	4	
		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	
		Sequence number	Ulong	4	
		Length of frame	Ulong	4	
		Base station health	Ulong	4	
variable	freq	Frequency indicator where 0 = L1 2 = L2 (1 is reserved for future use)	Ulong	4	variable
	smooth	Smoothing interval, see *Table 61, RTCM1819 Smoothing Interval on Page 205*	Ulong	4	
	GNSS time	GNSS time of measurement (μs)	Long	4	
	#obs	Number of observations with information to follow	Ulong	4	
variable	multi bit	Multiple message indicator	Ulong	4	variable
	code	Is code P Code? 0 = FALSE 1 = TRUE	Ulong	4	
	sat type	Satellite type 0 = GPS 1 = GLONASS	Ulong	4	
	prn	Satellite PRN number	Ulong	4	
	quality	Data quality indicator, see *Table 60, RTCM1819 Data Quality Indicator on Page 204*	Ulong	4	
	multipath	Multipath indicator, see *Table 62, RTCM1819 Multipath Indicator on Page 205*	Ulong	4	
	range	Pseudorange (2/100 m)	Ulong	Ulong	
variable...	Next RTCM19 observation offset = variable				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata1819a ontime 2

ASCII Example:

```
#RTCMDATA1819A,COM1,1,73.5,FINESTEERING,1117,161114.000,00100020,b077,399;
0,0,4523,0,0,6,
2,0,200000,11,
1,1,0,3,0,1,324771431,
1,1,0,15,0,1,64534978,
1,1,0,18,0,1,198055064,
1,1,0,21,0,1,426607534,
1,1,0,17,0,1,-101227879,
1,1,0,6,0,7,-70480075,
1,1,0,26,0,1,-205262773,
1,1,0,23,0,1,46251638,
1,1,0,28,0,1,167164502,
1,1,0,31,0,1,77539005,
1,1,0,22,0,3,-19,
0,0,4523,0,0,6,
2,0,200000,11,
1,1,0,3,2,3,1114597101,
1,1,0,15,2,3,999274497,
1,1,0,18,2,3,1022282623,
1,1,0,21,2,3,1151773907,
1,1,0,17,2,3,1015290815,
1,1,0,6,2,3,1207662688,
1,1,0,26,2,3,1085620069,
1,1,0,23,2,3,1029707897,
1,1,0,28,2,3,1240811844,
1,1,0,31,2,3,1242647691,
1,1,0,22,4,3,1241415667*820e5a7b
```

Table 60: RTCM1819 Data Quality Indicator

Code	Pseudorange Error
0	≤ 0.020 m
1	≤ 0.030 m
2	≤ 0.045 m
3	≤ 0.066 m
4	≤ 0.099 m
5	≤ 0.148 m
6	≤ 0.220 m
7	≤ 0.329 m
8	≤ 0.491 m
9	≤ 0.732 m
10	≤ 1.092 m
11	≤ 1.629 m
12	≤ 2.430 m
13	≤ 3.625 m
14	≤ 5.409 m
15	> 5.409 m

Table 61: RTCM1819 Smoothing Interval

Code	Smoothing Interval (Minutes)
0	0 to 1
1	1 to 5
2	5 to 15
3	Undefined smoothing interval

Table 62: RTCM1819 Multipath Indicator

Code	Multipath Error
0	≤ 0.100 m
1	≤ 0.149 m
2	≤ 0.223 m
3	≤ 0.332 m
4	≤ 0.495 m
5	≤ 0.739 m
6	≤ 1.102 m
7	≤ 1.644 m
8	≤ 2.453 m
9	≤ 3.660 m
10	≤ 5.460 m
11	≤ 8.145 m
12	≤ 12.151 m
13	≤ 18.127 m
14	> 18.127 m
15	Undetermined multipath

3.4.65 RTCMDATA2021 Measurement Corrections *RTK*

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 400**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header (for RTCM20)	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	freq	Frequency indicator 0 = L1 2 = L2	Ulong	4	H+24
9	Reserved		Ulong	4	H+28
10	GNSS time	Global Navigation Satellite System (GNSS) time of measurement (μs)	Long	4	H+32
11	#obs	Number of observation with information to follow	Long	4	H+36
12	multi bit	Multiple message indicator	Ulong	4	H+40
13	code	Is code P Code? 0 = FALSE 1 = TRUE	Ulong	4	H+44
14	sat type	Satellite type 0 = GPS 1 = GLONASS	Ulong	4	H+48
15	prn	Satellite PRN number	Ulong	4	H+52
16	quality	Data quality indicator, see *Table 63, RTCM2021 Data Quality Indicator on Page 208*	Ulong	4	H+56
17	continuity	Cumulative loss of continuity indicator with a loss of lock counter	Ulong	4	H+60
18	IODE	Issue of ephemeris data	Ulong	4	H+64
19	phase	Carrier phase correction (1/256 cycles)	Long	4	H+68
20...	Next RTMC20 observation offset = H+40 + (#obs x 32)				

variable	RTCM header (for RTCM21)	RTCM message type	Ulong	4	variable
		Base station ID	Ulong	4	
		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	
		Sequence number	Ulong	4	
		Length of frame	Ulong	4	
		Base station health	Ulong	4	
variable	freq	Frequency indicator	Ulong	4	variable
	Reserved		Ulong	4	
	GNSS time	GNSS time of measurement	Long	4	
	#obs	Number of observations with information to follow	Ulong	4	
variable	rate scale	Pseudorange rate correction scale factor 0 = 0.002 1 = 0.032	Ulong	4	variable
	code	Is code P Code? 0 = FALSE 1 = TRUE	Ulong	4	
	sat type	Satellite type 0 = GPS 1 = GLONASS	Ulong	4	
	prn	Satellite PRN number	Ulong	4	
	corr scale	Pseudorange correction scale factor 0 = 0.02 1 = 0.32	Ulong	4	
	quality	Data quality indicator, see *Table 63, RTCM2021 Data Quality Indicator on Page 208*	Ulong	4	
	multipath	Multipath indicator, see *Table 64, RTCM2021 Multipath Indicator on Page 208*	Ulong	4	
	IODE	Issue of ephemeris data	Ulong	4	
	range corr	Pseudorange correction (scaled)	Long	4	
	range rate	Pseudorange range correction rate (scaled)	Long	4	
variable...	Next RTCM21 observation offset = variable				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata2021a ontime 10

ASCII Example:

```
#RTCMDATA2021A,COM1,0,72.0,FINESTEERING,1117,161400.000,00100020,fc4d,399;
0,0,5000,0,0,6,
2,0,0,10,
0,1,0,3,0,1,43,-324,
0,1,0,15,0,1,96,-812,
0,1,0,18,0,1,1,514,
0,1,0,21,0,1,153,997,
0,1,0,6,0,7,88,-779,
0,1,0,26,0,1,35,39,
0,1,0,23,0,1,167,229,
0,1,0,28,0,1,22,1738,
0,1,0,31,0,1,125,5194,
0,1,0,22,0,4,27,-102,
0,0,5000,0,0,6,
2,0,0,10,
0,1,0,3,0,0,3,43,-661,-9,
0,1,0,15,0,0,3,96,-479,-11,
0,1,0,18,0,0,3,1,-152,-8,
0,1,0,21,0,0,3,153,-933,-9,
0,1,0,6,0,0,3,88,-2151,-12,
0,1,0,26,0,0,3,35,-630,-8,
0,1,0,23,0,0,3,167,-259,-10,
0,1,0,28,0,0,3,22,-1503,-7,
0,1,0,31,0,0,3,125,-1905,-9,
0,1,0,22,0,0,3,27,-2281,-14*f3963d96
```

Table 63: RTCM2021 Data Quality Indicator

Code	Pseudorange Error
0	≤ 0.1 m
1	≤ 0.25 m
2	≤ 0.5 m
3	≤ 1.0 m
4	≤ 2.0 m
5	≤ 3.5 m
6	≤ 5 m
7	> 5

Table 64: RTCM2021 Multipath Indicator

Code	Multipath Error
0	≤ 0.1 m
1	≤ 0.25 m
2	≤ 0.5 m
3	≤ 1.0 m
4	≤ 2.5 m
5	≤ 5 m
6	> 5 m
7	Undetermined multipath

3.4.66 RTCMDATA22 Extended Base Station RTK

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 401**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	L1 ECEF-X	L1 ECEF ΔX correction (1/256 cm)	Long	4	H+24
9	L1 ECEF-Y	L1 ECEF ΔY correction (1/256 cm)	Long	4	H+28
10	L1 ECEF-Z	L1 ECEF ΔZ correction (1/256 cm)	Long	4	H+32
11	Reserved		Ulong	4	H+36
12	height stat	No height flag where 0 = FALSE 1 = TRUE	Enum	4	H+40
13	phase center	Antenna L1 phase center height (1/256 cm)	Ulong	4	H+44
14	L2 ECEF-X	L1 ECEF ΔX correction (1/256 cm)	Long	4	H+48
15	L2 ECEF-Y	L1 ECEF ΔY correction (1/256 cm)	Long	4	H+52
16	L2 ECEF-Z	L1 ECEF ΔZ correction (1/256 cm)	Long	4	H+56
17	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+60
18	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtcmdata22a ontime 10

ASCII Example:

```
#RTCMDATA22A,COM1,0,70.0,FINESTEERING,1117,161590.000,00100020,990f,399;
1730644,0,5316,2324476,8451556,6,
61,-64,120,0,TRUE,0,0,0,0*b86ebf12
```

3.4.67 RTCMDATA59 Type 59N-0 NovAtel RT20 Differential *RTK*

See the chapter on *Message Formats* in *Volume 1* of this manual for information on RTCM standard logs.

Structure: **Message ID: 403**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	RTCM header	RTCM message type	Ulong	4	H
3		Base station ID	Ulong	4	H+4
4		Modified Z count where the Z count week number is the week number from subframe 1 of the ephemeris	Ulong	4	H+8
5		Sequence number	Ulong	4	H+12
6		Length of frame	Ulong	4	H+16
7		Base station health	Ulong	4	H+20
8	subtype	Message subtype	Char	4[a]	H+24
9	min psr	Minimum pseudorange (m)	Long	4	H+28
10	time offset	Time difference between the Z-count time and the measurement time where Z-count time from subframe 1 of the ephemeris (0.1 seconds / lsb)	Long	4	H+32
10	Reserved		Ulong	4	H+36
11	#prn	Number of PRNs with information to follow	Ulong	4	H+40
12	prn	Satellite PRN number	Ulong	4	H+44
13	lock	Lock time: 0 = <20 seconds 1 = 20-40 seconds 2 = 40-80 seconds 3 = >80 seconds	Ulong	4	H+48
14	psr	Pseudorange correction (1/10 m)	Ulong	4	H+52
15	adr	Accumulated Doppler Range (ADR) correction (1/1000 m)	Long	4	H+56
16...	Next PRN offset = H+44 + (#prns x 16)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 3 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log rtcmdata59a ontime 10

ASCII Example:

```
#RTCMDATA59A,COM1,0,71.5,FINESTEERING,1117,323592.000,00140000,3df8,337;
67108864,0,5320,67272710,0,6,
78,20341249,0,0,10,
15,2,36613566,-153,
30,2,24667890,-209,
17,3,21548029,-138,
6,3,6,-68,
23,3,43118232,-225,
5,0,45608604,-118,
24,3,31489783,-218,
10,3,5398457,55,
22,3,35679766,33,
26,2,42925557,-101*203b6b3d
```

3.4.68 RTKDATA RTK Solution Parameters RTK

This is the "RTK output" log, and it contains miscellaneous information regarding the RTK solution. It is based on the matched update. Note that the length of the log messages will vary depending on the number of common satellites (on both rover and base stations) in the solution, a quantity represented by *#sv* in the field numbers.

See also the BESTPOS log (the best available position computed by one receiver) and the MATCHEDPOS log (positions that have been computed from time matched base and rover observations), on *Pages 122 and 157* respectively.

Structure: **Message ID: 215**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status (see *Table 40, Solution Status on Page 124*)	Enum	4	H
3	pos type	Position type (see *Table 39, Position or Velocity Type on Page 123*)	Enum	4	H+4
4	rtk info	RTK information (see *Table 67, RTK Information on Page 213*)	Ulong	4	H+8
5	#obs	Number of observations tracked	Uchar	1	H+12
6	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+13
7	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+14
8	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+15
9	Reserved		Uchar	1	H+16
10			Uchar	1	H+17
11			Uchar	1	H+18
12			Uchar	1	H+19
13	search stat	Searcher status (see *Table 65 on Page 213*).	Enum	4	H+20
14	# lane	Number of possible lane combinations.	Ulong	4	H+24
15-23	[C]	The C_{xx},C_{xy},C_{xz},C_{yx},C_{yy},C_{yz},C_{zx},C_{zy} and C_{zz} components in $(meters)^2$, of the ECEF position covariance matrix (3x3).	Float	36	H+28
24	Δ x	Float solution baseline in ECEF - x.	Double	8	H+64
25	Δ y	Float solution baseline in ECEF - y.	Double	8	H+72
26	Δ z	Float solution baseline in ECEF - z.	Double	8	H+80
27	x σ	Standard deviation of float solution baseline - x (m).	Float	4	H+88
28	y σ	Standard deviation of float solution baseline - y (m).	Float	4	H+92
29	z σ	Standard deviation of float solution baseline - z (m).	Float	4	H+96
30	ref PRN	Base PRN.	Ulong	4	H+100
31	# SV	Number of SVs to follow.	Long	4	H+104
32	PRN	Satellite PRN number of range measurement.	Ulong	4	H+108
33	amb	Ambiguity type (see *Table 66 on Page 213*).	Enum	4	H+112
34	res	Residual (m).	Float	4	H+116
35...	Next SV offset = H + 108 + (obs x 12)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+108+(12x obs)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtkdataa onchanged

ASCII Example:

```
#RTKDATAA,COM1,0,49.0,FINESTEERING,1037,507891.000,00000028,d89c,0;
SOL_COMPUTED,L1_FLOAT,00000107,10,10,10,0,0,0,0,0,COMPLETE,1,0.000006136,
0.000003797,-0.000003287,0.0000003797,0.000013211,-0.000007043,-0.000006287,
-0.000007043,0.000018575,3.2209,-3.0537,-1.2024,0.0183,0.0138,0.0124,1,7,
21,6,-0.001199,0.000,
...
25,6,-0.004366,0.008*481a9f30
```

Table 65: Searcher Type

Searcher Type (binary)	Searcher Type (ASCII)	Description
0	NONE_REQUESTED	No search requested.
1	BUFFERING_MEASUREMENTS	Buffering measurements.
2	SEARCHING	Currently searching.
3	COMPLETE	Searcher made decision.
4	HANDOFF_COMPLETE	Hand off to L1 and L2 complete.

Table 66: Ambiguity Types

Ambiguity Type (binary)	Ambiguity Type (ASCII)	Description
0	UNDEFINED	Undefined ambiguity
1	L1_FLOAT	Floating L1 ambiguity
2	IONOFREE_FLOAT	Floating ionospheric-free ambiguity
3	NARROW_FLOAT	Floating narrow-lane ambiguity.
4	NLF_FROM_WL1	Floating narrow-lane ambiguity derived from integer wide-lane ambiguity
5	L1_INT	Integer L1 ambiguity
6	WIDE_INT	Integer wide-lane ambiguity
7	NARROW_INT	Integer narrow-lane ambiguity
8	IONOFREE_DISCRETE	Discrete ionospheric-free ambiguity

Table 67: RTK Information

Bit #	Mask	Description	Bit = 0	Bit = 1
0	0x00000001	RTK dynamics	Static	Dynamic
1	0x00000002	RTK dynamics mode	Auto	Forced
2	0x00000004	Severe differential ionosphere detected	No	Yes
3-31	0xFFFFFF8	Reserved		

3.4.69 RTKPOS RTK Low Latency Position Data *RTK*

This log contains the low latency RTK position computed by the receiver, along with two status flags. In addition, it reports other status indicators, including differential age, which is useful in predicting anomalous behavior brought about by outages in differential corrections. This log is recommended for kinematic operation. Better accuracy can be obtained in static operation with the MATCHEDPOS log.

With the system operating in an RTK mode, this log will reflect if the solution is a good RTK low latency solution (from extrapolated base station measurements) or invalid. A valid RTK low latency solution will be computed for up to 30 seconds after reception of the last base station observation. The degradation in accuracy, due to differential age, is reflected in the standard deviation fields, and is summarized in the *Differential Positioning* section in the *GPS Overview* appendix in *Volume 1* of this manual.

Structure: **Message ID: 141**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status (see *Table 40, Solution Status on Page 124*)	Enum	4	H
3	pos type	Position type (see *Table 39, Position or Velocity Type on Page 123*)	Enum	4	H+4
4	lat	Latitude	Double	8	H+8
5	lon	Longitude	Double	8	H+16
6	hgt	Height above mean sea level	Double	8	H+24
7	undulation	Undulation	Float	4	H+32
8	datum id#	Datum ID number (see *Chapter 2, Table 17, Datum Transformation Parameters on Page 54*)	Enum	4	H+36
9	lat σ	Latitude standard deviation	Float	4	H+40
10	lon σ	Longitude standard deviation	Float	4	H+44
11	hgt σ	Height standard deviation	Float	4	H+48
12	stn id	Base station ID	Char[4]	4	H+52
13	diff_age	Differential age	Float	4	H+56
14	sol_age	Solution age in seconds	Float	4	H+60
15	#obs	Number of observations tracked	Uchar	1	H+64
16	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+65
17	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+66
18	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+67
19	Reserved		Uchar	1	H+68
20			Uchar	1	H+69
21			Uchar	1	H+70
22			Uchar	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtkposa ontime 1

ASCII Example:

```
#RTKPOSA,COM1,0,49.0,FINESTEERING,1037,507891.000,00000028,d89c,0;
SOL_COMPUTED,L1_FLOAT,51.11638130574,-114.03829239121,1048.2656,-16.271239716,
61,0.0472,0.0659,0.0430,"0",1.0,60.000,10,10,10,0,0,0,0,0*481a9f30
```

3.4.70 RTKVEL RTK Velocity RTK

This log contains the RTK velocity information computed by the receiver. In addition, it reports a velocity status indicator, which is useful in indicating whether or not the corresponding data is valid and differential age, which is useful in predicting anomalous behavior brought about by outages in differential corrections. The velocity measurements sometimes have a latency associated with them. The time of validity is the time tag in the log minus the latency value. See also the table footnote for velocity logs on *Page 107*.

With the system operating in an RTK mode, this log will reflect if the solution is a good RTK Low Latency solution (from extrapolated base station measurements) or invalid. A valid RTK Low Latency solution will be computed for up to 30 seconds after reception of the last base station observation. The degradation in accuracy due to differential age is reflected in the standard deviation fields, and is summarized in the *Differential Positioning* section in the *GPS Overview* appendix in *Volume 1* of this manual.

The velocity is computed from consecutive RTK low latency updates. As such, it is an average velocity based on the time difference between successive position computations and not an instantaneous velocity at the RTKVEL time tag. The velocity latency to be subtracted from the time tag will normally be 1/2 the time between filter updates. Under default operation, the RTK low latency filter is updated at a rate of 2 Hz. This translates into a velocity latency of 0.25 seconds. The latency can be reduced by increasing the update rate of the RTK low latency filter by requesting the BESTVEL, RTKVEL, BESTPOS or RTKPOS messages at a rate higher than 2 Hz. For example, a logging rate of 10 Hz would reduce the velocity latency to 0.005 seconds. For integration purposes, the velocity latency should be applied to the record time tag.

Structure: **Message ID: 216**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	latency	A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results.	Float	4	H+8
5	age	Differential age	Float	4	H+12
6	hor spd	Horizontal speed over ground, in meters per second	Double	8	H+16
7	trk gnd	Actual direction of motion over ground (track over ground) with respect to True North, in degrees	Double	8	H+24
8	vert spd	Vertical speed, in meters per second, where positive values indicate increasing altitude (up) and negative values indicate decreasing altitude (down)	Double	8	H+32
9	Reserved		Float	4	H+40
10	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
11	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtkvela ontime 1

ASCII Example:

```
#RTKVELA,COM1,0,56.0,FINESTEERING,1037,322331.500,00000028,c9ad,0;SOL_COMPUTED
,DOPPLER_VELOCITY,0.250,1.000,0.112280259,114.237364983,0.1229,0.0659*9f5e535d
```

3.4.71 RTKXYZ RTK Cartesian Position and Velocity RTK

This log contains the receiver's low latency position and velocity in ECEF coordinates. The position and velocity status field's indicate whether or not the corresponding data is valid. See *Figure 6, Page 127* for a definition of the ECEF coordinates.

The velocity measurements sometimes have a latency associated with them. The time of validity is the time tag in the log minus the latency value.

With the system operating in an RTK mode, this log will reflect if the solution is a good RTK Low Latency solution (from extrapolated base station measurements) or invalid. A valid RTK Low Latency solution will be computed for up to 30 seconds after reception of the last base station observation. The degradation in accuracy due to differential age is reflected in the standard deviation fields, and is summarized in the *Differential Positioning* section in the *GPS Overview* appendix in *Volume 1* of this manual.

The velocity is computed from consecutive RTK low latency updates. As such, it is an average velocity based on the time difference between successive position computations and not an instantaneous velocity at the RTKVEL time tag. The velocity latency to be subtracted from the time tag will normally be 1/2 the time between filter updates. Under default operation, the RTK low latency filter is updated at a rate of 2 Hz. This translates into a velocity latency of 0.25 seconds. The latency can be reduced by increasing the update rate of the RTK low latency filter by requesting the BESTXYZ message at a rate higher than 2 Hz. For example, a logging rate of 10 Hz would reduce the velocity latency to 0.005 seconds. For integration purposes, the velocity latency should be applied to the record time tag

See also the BESTXYZ and MATCHEDXYZ logs, on *Pages 126* and *161* respectively.

Structure: **Message ID: 244**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	P-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H
3	pos type	Position type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+4
4	P-X	Position X-coordinate (m)	Double	8	H+8
5	P-Y	Position Y-coordinate (m)	Double	8	H+16
6	P-Z	Position Z-coordinate (m)	Double	8	H+24
7	P-X σ	Standard deviation of P-X (m)	Float	4	H+32
8	P-Y σ	Standard deviation of P-Y (m)	Float	4	H+36
9	P-Z σ	Standard deviation of P-Z (m)	Float	4	H+40
10	V-sol status	Solution status, see *Table 40, Solution Status on Page 124*	Enum	4	H+44
11	vel type	Velocity type, see *Table 39, Position or Velocity Type on Page 123*	Enum	4	H+48
12	V-X	Velocity vector along X-axis (m)	Double	8	H+52
13	V-Y	Velocity vector along Y-axis (m)	Double	8	H+60

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
14	V-Z	Velocity vector along Z-axis (m)	Double	8	H+68
15	V-X σ	Standard deviation of V-X (m)	Float	4	H+76
16	V-Y σ	Standard deviation of V-Y (m)	Float	4	H+80
17	V-Z σ	Standard deviation of V-Z (m)	Float	4	H+84
18	stn ID	Base station identification	Char[4]	4	H+88
19	V-latency	A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results.	Float	4	H+92
20	diff_age	Differential age	Float	4	H+96
21	sol_age	Solution age in seconds	Float	4	H+100
22	#obs	Number of observations tracked	Uchar	1	H+104
23	#GPSL1	Number of GPS L1 ranges used in computation	Uchar	1	H+105
24	#L1	Number of GPS L1 ranges above the RTK mask angle	Uchar	1	H+106
25	#L2	Number of GPS L2 ranges above the RTK mask angle	Uchar	1	H+107
26	Reserved		Char	1	H+108
27			Char	1	H+109
28			Char	1	H+110
29			Char	1	H+111
30	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+112
31	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rtkxyza ontime 1

ASCII Example:

```
#RTKXYZA,COM1,0,56.5,FINESTEERING,1058,410358.000,00000028,aec2,37;
SOL_COMPUTED,L1_INT,-1634527.490,-3664605.319,4942486.969,0.0121,0.0150,
0.0265,SOL_COMPUTED,L1_INT,0.006,0.007,-0.005,0.0000,0.0000,0.0000,"AAAA",
0.250,2.000,0.000,9,6,6,6,0,0,0,0*320e2911
```

3.4.72 RXCONFIG Receiver Configuration

This log is used to output a list of all current command settings. When requested, an RXCONFIG log is output for each setting. See also the LOGLIST log on *Page 155* for a list of currently active logs.

Structure: **Message ID: 128**
Log Type: Polled

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header	-	H	0
2	e header	Embedded header	-	h	H
3	e msg	Embedded message	Varied	a	H + h
4	e xxxx	Embedded (inverted) 32-bit CRC (ASCII and Binary only). The embedded CRC is inverted so that the card will not recognize the embedded messages as messages to be output but will continue with the RXCONFIG message. If you wish to use the messages output from the RXCONFIG log, simply flip the embedded CRC around for individual messages.	Long	4	H+ h + a
5	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+ h + a + 4
6...	Next Log header offset = # of log headers to follow x (H + h + a + 4)				
7	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rxconfiga once

ASCII Example[1]:

```
#RXCONFIGA,COM1,23,61.0,UNKNOWN,0,7.144,00000028,898f,0;#ANTENNAPOWERA,COM1,2
3,61.0,UNKNOWN,0,7.144,00000028,898f,0;ON*182ae681*2ed9fe08
#RXCONFIGA,COM1,22,61.0,UNKNOWN,0,7.167,00000028,898f,0;#CLOCKADJUSTA,COM1,22
,61.0,UNKNOWN,0,7.167,00000028,898f,0;ENABLE*34c38b53*7a597498
#RXCONFIGA,COM1,21,61.0,UNKNOWN,0,7.180,00000028,898f,0;#CLOCKTYPEA,COM1,21,6
1.0,UNKNOWN,0,7.180,00000028,898f,0;TCXO,0.0000000000000000,0.000000000,0.000
000000*c9788294*a1ecf566
...
#RXCONFIGA,COM1,15,61.0,UNKNOWN,0,7.289,00000028,898f,0;#FIXA,COM1,15,61.0,UN
KNOWN,0,7.289,00000028,898f,0;NONE,-10000.000000,-10000.000000,
-10000.000000*d99c1c73*cd0edd08
#RXCONFIGA,COM1,14,61.0,FINESTEERING,1037,498794.357,00000028,898f,0;#INTERFA
CEMODEA,COM1,14,61.0,FINESTEERING,1037,498794.357,00000028,898f,0;COM2,RTCM,N
OVATEL,ON*dec71f28*5debdc5e
...
#RXCONFIGA,COM1,0,61.0,FINESTEERING,1037,498737.616,00000028,898f,0;#LOGA,COM
1,0,61.0,FINESTEERING,1037,498737.616,00000028,898f,0;COM1_4,GPSOL_POSB,ONTIM
E,1.000000,0.000000,HOLD*eb2fe18b*23604814
```

1. The embedded CRCs are flipped to make the embedded messages recognizable to the receiver. For example, consider the first embedded message above.

182ae681: 00011000001010101110011010000001
11100111110101010001100101111110: e7d5197e

Its CRC is really e7d5197e.

3.4.73 RXHWLEVELS Receiver Hardware Levels

This log contains the receiver environmental and voltage parameters.

Structure: **Message ID: 195**
Log Type: Polled

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	temp	Board temperature (degrees celsius)	Float	4	H
3	ant current	Approximate internal antenna current (A)	Float	4	H+4
4	core volt	CPU core voltage (V)	Float	4	H+8
5	supply volt	Receiver supply voltage (V)	Float	4	H+12
6	rf volt	5V RF supply voltage (V)	Float	4	H+16
7	int lna volt	Internal LNA voltage level (V)	Float	4	H+20
8	GPAI	General purpose analog input (V)	Float	4	H+24
9	Reserved		Float	4	H+28
10			Float	4	H+32
11	lna volt	LNA voltage (V) at card output	Float	4	H+36
12	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+40
13	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rxhwlevelsa ontime 60

Table 68: Receiver Hardware Parameters [a]

	Temp.	Antenna Current	Core Voltage	Supply Voltage	RF Voltage	Internal LNA Voltage	GPAI	LNA Voltage
Min	-40	0	1.42	OEM4: 6 Euro4: 4.9	4.75	4.75	0	0
Max	100 [b]	100	1.58	OEM4: 18 Euro4: 5.2	5.25	5.25	2.75	30
Typical	40	40	1.5	OEM4: 12 Euro4: 5	5	5	0	5

a. The values in this table are the same for the OEM4 and Euro4 platform, except for the supply voltage. Check the RXHWLEVEL syntax table for the units of each field.

b. The board temperature is about 15°C higher than the ambient temperature. Bit 1, in *Table 70, Receiver Status on Page 223*, turns on as a warning when the board temperature is above 100°C and a hazardous temperature error message is generated at 110°C.

ASCII Example:

```
#RXHWLEVELSA,COM1,0,62.0,FINESTEERING,1114,237777.536,00000020,01a6,209;
26.609,-0.003,1.572,8.486,4.968,4.967,0.000,0.000,0.000,0.000*d8f899cb
```

3.4.74 RXSTATUS Receiver Status

This log conveys various status parameters of the GPS receiver system. These include the Receiver Status and Error words which contain several flags specifying status and error conditions. If an error occurs (shown in the Receiver Error word) the receiver will idle all channels, turn off the antenna, and disable the RF hardware as these conditions are considered to be fatal errors. The log contains a variable number of status words to allow for maximum flexibility and future expansion.

The receiver gives the user the ability to determine the importance of the status bits. In the case of the Receiver Status, setting a bit in the priority mask will cause the condition to trigger an error. This will cause the GPSCard to idle all channels, turn off the antenna, and disable the RF hardware, the same as if a bit in the Receiver Error word is set. Setting a bit in an Auxiliary Status priority mask will cause that condition to set the bit in the Receiver Status word corresponding to that Auxiliary Status.

Receiver Errors automatically generate event messages. These event messages are output in RXSTATUSEVENT logs. It is also possible to have status conditions trigger event messages to be generated by the receiver. This is done by setting/clearing the appropriate bits in the event set/clear masks. The set mask tells the receiver to generate an event message when the bit becomes set. Likewise, the clear mask causes messages to be generated when a bit is cleared. See the STATUSCONFIG command on *Page 97* for details.

If you wish to disable all these messages without changing the bits, simply UNLOG the RXSTATUSEVENT logs on the appropriate ports.

Note that Field #4, the receiver status word as represented in *Table 70*, is also in Field #8 of the header. See the *ASCII Example* and *Table 70* on *Page 223* for clarification.

✉ Refer also to the chapter on *Built-In Status Tests* in *Volume 1* of this manual.

Structure: **Message ID: 93**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	error	Receiver error (see *Table 69 on Page 222*). A value of zero indicates no errors.	ULong	4	H
3	# stats	Number of status codes (including Receiver Status). Normally = 4	ULong	4	H+4
4	rxstat	Receiver status word (see *Table 70 on Page 223*)	ULong	4	H+8
5	rxstat pri	Receiver status priority mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+12
6	rxstat set	Receiver status event set mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+16
7	rxstat clear	Receiver status event clear mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+20
8	aux1stat	Auxiliary 1 status word (see *Table 71 on Page 224*)	ULong	4	H+24

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
9	aux1stat pri	Auxiliary 1 status priority mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+28
10	aux1stat set	Auxiliary 1 status event set mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+32
11	aux1stat clear	Auxiliary 1 status event clear mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+36
12	aux2stat	Auxiliary 2 status word (see *Table 72 on Page 224*)	ULong	4	H+40
13	aux2stat pri	Auxiliary 2 status priority mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+44
14	aux2stat set	Auxiliary 2 status event set mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+48
15	aux2stat clear	Auxiliary 2 status event clear mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+52
16	aux3stat	Auxiliary 3 status word (see *Table 73 on Page 224*)	ULong	4	H+56
17	aux3stat pri	Auxiliary 3 status priority mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+60
18	aux3stat set	Auxiliary 3 status event set mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+64
19	aux3stat clear	Auxiliary 3 status event clear mask, which can be set using the STATUSCONFIG command (*see Page 97)*	ULong	4	H+68
20...	Next status code offset = H + 8 + (#stats x 64)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+8+(#stats x 64)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rxstatusa onchanged

ASCII Example:

```
#RXSTATUSA,COM1,0,54.5,FINESTEERING,1075,321753.938,00040028,bfad,97;
00000000,4,00040028,00000000,00000000,00000000,00400006,00000008,00000000,
00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000000,
00000000*4ef4a6c3
```

Table 69: Receiver Error

Nibble #	Bit #	Mask	Description	Bit = 0	Bit = 1
N0	0	0x00000001	Dynamic Random Access Memory (DRAM) status	OK	Error
	1	0x00000002	No application	OK	Error
N1	4	0x00000010	Electronic Serial Number (ESN) access status	OK	Error
	5	0x00000020	Authorization code status	OK	Error
	6	0x00000040	Slow ADC status	OK	Error
	7	0x00000080	Supply voltage status	OK	Error
N2	8	0x00000100	Thermometer status	OK	Error
	9	0x00000200	Hazardous temperature	OK	Error
	10	0x00000400	MINOS4 status	OK	Error
	11	0x00000800	PLL RF1 hardware status - L1	OK	Error
N3	12	0x00001000	PLL RF2 hardware status - L2	OK	Error
	13	0x00002000	RF1 hardware status - L1	OK	Error
	14	0x00004000	RF2 hardware status - L2	OK	Error
	15	0x00008000	NVM status	OK	Error
N4	16	0x00010000	Reserved	OK	Error
	17	0x00020000		OK	Error
	18	0x00040000		OK	Error
	19	0x00080000		OK	Error

Table 70: Receiver Status

Nibble #	Bit #	Mask	Description	Bit = 0	Bit = 1
N0	0	0x00000001	Error Occurred, see *Table 69, Receiver Error on Page 222*	No Error	Error
	1	0x00000002	Temperature	Within Specifications	Warning
	2	0x00000004	Voltage Supply Status	Good	Warning
	3	0x00000008	Antenna Not Powered, see *ANTENNAPOWER* on *Page 38*	Powered	Not Powered
N1	4	0x00000010	LNA Status Failed	OK	Fail
	5	0x00000020	Antenna Open	Good	Open
	6	0x00000040	Antenna Shorted	Good	Shorted
	7	0x00000080	CPU Overload	No Overload	Overload
N2	8	0x00000100	COM1 Buffer Overrun	Good	Overrun
	9	0x00000200	COM2 Buffer Overrun	Good	Overrun
	10	0x00000400	COM3 Buffer Overrun	Good	Overrun
	11	0x00000800	Reserved		
N3	12	0x00001000			
	13	0x00002000			
	14	0x00004000	RF1 Jammed	Good	Jammed
	15	0x00008000	RF1 AGC Bad	Good	Bad
N4	16	0x00010000	RF2 Jammed	Good	Jammed
	17	0x00020000	RF2 AGC Bad	Good	Bad
	18	0x00040000	Almanac Invalid	Valid	Invalid
	19	0x00080000	Position Solution Invalid	Valid	Invalid
N5	20	0x00100000	Fixed Position, see *FIX* on *Page 65*	Not Fixed	Fixed
	21	0x00200000	Clock Steering Disabled	Enabled	Disabled
	22	0x00400000	Clock Model Invalid	Valid	Invalid
	23	0x00800000	Euro4 External Oscillator	PLL Not Locked	PLL Locked
N6	24	0x01000000	Reserved		
	25	0x02000000			
	26	0x04000000			
	27	0x08000000			
N7	28	0x10000000			
	29	0x20000000			
	30	0x40000000	AUX2 Status Event	No Event	Event
	31	0x80000000	AUX1 Status Event	No Event	Event

Table 71: Auxiliary 1 Status

Nibble #	Bit #	Mask	Description	Bit = 0	Bit = 1
N0	0	0x00000001	COM1 connected	Connected	Not connected
	1	0x00000002	COM2 connected	Connected	Not connected
	2	0x00000004	COM3 connected	Connected	Not connected
	3	0x00000008	Position Averaging	Off	On

Table 72: Auxiliary 2 Status

Nibble #	Bit #	Mask	Description	Bit = 0	Bit = 1
N0	0	0x0000001	Reserved		

Table 73: Auxiliary 3 Status

Nibble #	Bit #	Mask	Description	Bit = 0	Bit = 1
N0	0	0x0000001	Reserved		

3.4.75 RXSTATUSEVENT Status Event Indicator

This log is used to output event messages as indicated in the RXSTATUS log.

On startup, the receiver is set to log the RXSTATUSEVENTA log ONNEW on all ports. You can remove this message by using the UNLOG command.

✉ See also the chapter on *Built-In Status Tests* in *Volume 1* of this manual.

Structure: **Message ID: 94**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	word	The status word that generated the event message (see *Table 74, below*)	Enum	4	H
3	bit position	Location of the bit in the status word (see *Table 70, Receiver Status on Page 223*)	Ulong	4	H+4
4	event	Event type (see *Table 75, below*)	Enum	4	H+8
3	description	This is a text description of the event or error	Char[32]	32	H+12
5	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
6	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log rxstatuseventa onchanged

ASCII Example:

```
#RXSTATUSEVENTA,COM1,0,0.0,FINESTEERING,1037,508871.517,00240028,ff0c,0;
STATUS,21,SET,"Clock Model Invalid"*beae5676
```

Table 74: Status Word

Word (binary)	Word (ASCII)	Description
0	ERROR	Receiver Error word
1	STATUS	Receiver Status word
2	AUX1	Auxiliary 1 Status word
3	AUX2	Auxiliary 2 Status word
4	AUX3	Auxiliary 3 Status word

Table 75: Event Type

Event (binary)	Event (ASCII)	Description
0	CLEAR	Bit was cleared
1	SET	Bit was set

3.4.76 SATVIS Satellite Visibility

Satellite visibility log with additional satellite information.

Structure: **Message ID: 48**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sat vis	Is satellite visibility valid? 1 = TRUE 0 = FALSE	Enum	4	H
3	comp alm	Was complete almanac used? 1 = TRUE 0 = FALSE	Enum	4	H+4
4	#sat	Number of satellites with information to follow	Ulong	4	H+8
5	PRN	GPS satellite PRN number of range measurement.	Short	2	H+12
6	Reserved		Short	2	H+14
7	health	Satellite health[a]	Ulong	4	H+16
8	elev	Elevation (degrees)	Double	8	H+20
9	az	Azimuth (degrees)	Double	8	H+28
10	true dop	Theoretical Doppler of satellite	Double	8	H+36
11	app dop	Apparent Doppler for this GPSCard	Double	8	H+44
12	Next satellite offset = H + 12 + (#sat x 40)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+12+ (#sat x 40)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. Satellite health values may be found in ICD-GPS-200. To obtain copies of ICD-GPS-200, see ARINC in the appendix on *Standards and References* in *Volume 1* of this manual.

Recommended Input:

log satvisa ontime 60

ASCII Example:

```
#SATVISA,COM1,0,44.0,FINESTEERING,1039,490308.000,00000028,6002,0;
TRUE,TRUE,27,
14,0,0,74.5,267.4,458.2,458.926672761,
25,0,0,61.3,73.7,-1252.6,-1251.902056196,
1,0,0,55.9,277.4,1799.2,1799.897879028,
16,0,0,33.6,305.9,2874.8,2875.534296744,
22,0,0,21.6,146.3,3450.4,3451.136917152,
30,0,0,15.8,38.6,-1911.6,-1910.870440955,
...
26,0,0,-82.0,114.8,-188.9,-188.237459086*bf8c9522
```

3.4.77 SATXYZ SV Position in ECEF Cartesian Coordinates

When combined with a RANGE log, this data set contains the decoded satellite information necessary to compute the solution: satellite coordinates (ECEF WGS84), satellite clock correction, ionospheric corrections and tropospheric corrections (Hopfield model). The corrections are to be added to the pseudoranges. Only those satellites that are healthy are reported here. See also, *Figure 6, The WGS84 ECEF Coordinate System on Page 127*.

Structure: **Message ID: 270**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	Reserved		Double	8	H
3	#sat	Number of satellites with Cartesian information to follow	Ulong	4	H+8
4	prn	Satellite PRN number	Ulong	4	H+12
5	x	Satellite X coordinates (ECEF, m)	Double	8	H+16
6	y	Satellite Y coordinates (ECEF, m)	Double	8	H+24
7	z	Satellite Z coordinates (ECEF, m)	Double	8	H+32
8	clk corr	Satellite clock correction (m)	Double	8	H+40
9	ion corr	Ionospheric correction (m)	Double	8	H+48
10	trop corr	Tropospheric correction (m)	Double	8	H+56
11	Reserved		Double	8	H+64
12			Double	8	H+72
13	Next satellite offset = H + 12 + (#sat x 68)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+12+ (#sat x 68)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log satxyz ontime 1

ASCII Example:

```
#SATXYZA,COM1,0,55.5,FINESTEERING,1075,321955.000,00040028,75fd,97;
0.0,5,
17,4912363.4478,-21873193.4180,13712406.6234,80726.382,7.627007266
,3.186367161,0.000000000,0.000000000,
3,-18466102.2956,4149634.2037,18615299.2261,9459.257,
7.638836266,3.336822112,0.000000000,0.000000000,
9,15809955.4029,-20380283.5629,5122711.0922,-1192.473,13.205660767,
10.519492770,0.000000000,0.000000000,
26,12760597.3855,8106208.6797,21767386.6663,161356.045,
8.973848676,4.427102154,0.000000000,0.000000000,
21,-20362669.2858,-10170553.3022,14511463.0224,4328.159,
7.840224374,3.224598285,0.000000000,0.000000000*a888df72
```

3.4.78 TIME Time Data

This log provides several time related pieces of information. These include receiver clock offset and UTC time and offset. It also reports any error in the 1PPS signal.

Structure: **Message ID: 101**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	clock status	Clock model status (not including current measurement data), see *Table 41 on Page 128*	Enum	4	H
3	offset	Receiver clock offset, in seconds from GPS time. A positive offset implies that the receiver clock is ahead of GPS time. To derive GPS time, use the following formula: GPS time = receiver time - offset	Double	8	H+4
4	offset std	Receiver clock offset standard deviation.	Double	8	H+12
5	utc offset	This field represents the offset of GPS time from UTC time, computed using almanac parameters. UTC time is GPS time plus the current UTC offset plus the receiver clock offset. UTC time = GPS time + offset + UTC offset	Double	8	H+20
6	utc year	UTC year	Ulong	4	H+28
7	utc month	UTC month (0-12)	Uchar	1	H+32
8	utc day	UTC day (0-31)	Uchar	1	H+33
9	utc hour	UTC hour (0-23)	Uchar	1	H+34
10	utc min	UTC minute (0-59)	Uchar	1	H+35
11	utc millisec	UTC millisecond (0-59999)	Ulong	4	H+36
12	utc status	UTC status 0 = Invalid 1 = Valid	Enum	4	H+40
13	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+44
14	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log timea ontime 1

ASCII Example:

```
#TIMEA,COM1,0,58.0,FINESTEERING,1049,248050.000,00000128,5c46,0;
VALID,-0.000000002,0.000001901,-13.00000000294,2000,2,15,20,53,
57011,Valid*d6f14d5d
```

3.4.79 TIMESYNC Synchronize Time Between GPS Receivers

The TIMESYNC log is used in conjunction with the ADJUST1PPS command, see *Page 35*, to synchronize the time between GPS receivers.

The time data embedded in this log represents the time of the most recent 1PPS signal. This log should be issued from a communications port within 200 ms, of the last 1PPS event. See *Figure 8, below*, for an illustration.

Refer also to the *Transfer Time Between Receivers* section in *Volume 1* of this manual.

Structure: **Message ID: 492**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	week	GPS week number.	Ulong	4	H
3	mseconds	Number of milliseconds into the GPS week.	Ulong	4	H+4
4	time status	GPS Time Status, see *Table 6, GPS Time Status on Page 20.*	Enum	4	H+8
5	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+12
6	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

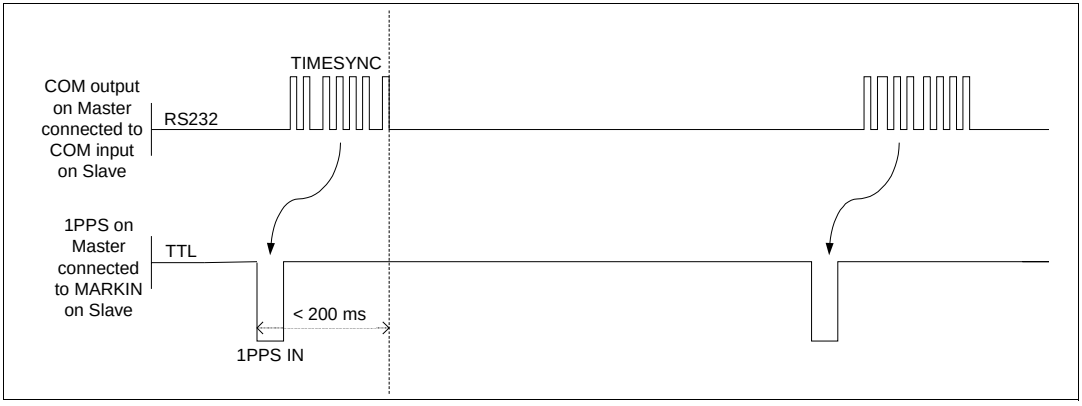


Figure 8: 1PPS Alignment-TIMESYNC Log

Recommended Input:

log timesync ontime 1

ASCII Example:

```
#TIMESYNCA,COM1,0,65.0,FINESTEERING,1058,272851.000,80000000,3a72,32800;
1058,272851000,FINESTEERING*a2f7b21b
```

3.4.80 TRACKSTAT Tracking Status

These logs provide channel tracking status information for each of the GPSCard parallel channels.

If both the L1 and L2 signals are being tracked for a given PRN, two entries with the same PRN will appear in the tracking status logs. As shown in *Channel Tracking Status on Page 177* these entries can be differentiated by bit 20, which is set if there are multiple observables for a given PRN, and bits 21-22, which denote whether the observation is for L1 or L2. This is to aid in parsing the data.

Structure: **Message ID: 83**
Log Type: Synch

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	sol status	Solution status (see *Table 40, Solution Status on Page 124*).	Enum	4	H
3	pos type	Position type (see *Table 39, Position or Velocity Type on Page 123*).	Enum	4	H+4
4	cutoff	Tracking elevation cut-off angle	Float	4	H+8
5	# chans	Number of hardware channels with information to follow	Long	4	H+12
6	PRN	GPS satellite PRN number of range measurement.	Short	2	H+16
7	Reserved		Short	2	H+18
8	ch-tr-status	Channel tracking status (see *Table 53, Channel Tracking Status on Page 177*)	ULong	4	H+20
9	psr	Pseudorange (m)	Double	8	H+24
10	Doppler	Doppler frequency (Hz)	Float	4	H+32
11	C/No	Carrier to noise density ratio (dB-Hz)	Float	4	H+36
12	locktime	Number of seconds of continuous tracking (no cycle slips)	Float	4	H+40
13	psr res	Pseudorange residual from pseudorange filter (m)	Float	4	H+44
14	reject	Range reject code from pseudorange filter (see *Table 76, Range Reject Codes on Page 231*)	Enum	4	H+48
15	psr weight	Pseudorange filter weighting	Float	4	H+52
16...	Next PRN offset = H + 16 + (#chans x 40)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+16+ (#chans x 40)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log trackstata ontime 1

ASCII Example:

```
#TRACKSTATA,COM1,0,64.5,FINESTEERING,1048,507426.700,00000028,228e,0;
SOL_COMPUTED,SINGLE,0.0,24,
5,0,8105c04,25052294.167,3388.063,40.188,63.560,7.209,GOOD,1.000,
…
2,0,1305c2b,21433408.821,1310.921,43.913,1514.040,0.000,OBSL2,1.000,
…
26,0,1305d6b,24444419.776,2365.217,31.530,1506.520,0.000,OBSL2,1.000*cd40e0aa
```

Table 76: Range Reject Codes

Reject Code (binary)	Reject Code (ASCII)	Description
0	GOOD	Observations are good
1	BADHEALTH	Bad satellite health is indicated by ephemeris data
2	OLDEPHEMERIS	Old ephemeris due to date not being updated during the last 3 hours
3	ECCENTRICANOMALY	Eccentric anomaly error during computation of the satellite's position
4	TRUEANOMALY	True anomaly error during computation of the satellite's position
5	SATCOORDINATEERROR	Satellite coordinate error during computation of the satellite's position
6	ELEVATIONERROR	Elevation error due to the satellite being below the cut-off angle
7	MISCLOSURE	Misclosure too large due to excessive gap between estimated and actual positions
8	NODIFFCORR	No compatible differential correction is available for this particular satellite
9	NOEPHEMERIS	Ephemeris data for this satellite has not yet been received
10	INVALIDIODE	Invalid IODE (Issue Of Data Ephemeris) due to mismatch between differential stations
11	LOCKEDOUT	Locked out: satellite is excluded by the user (LOCKOUT command)
12	LOWPOWER	Low power: satellite is rejected due to low carrier/noise ratio
13	OBSL2	L2 measurements are not being used by the filter
16	NOIONOCORR	No compatible ionospheric correction is available for this particular satellite
99	NA	No observation (a reject code is not applicable)

3.4.81 VALIDMODELS Valid Model Information

This log gives a list of valid authorized models available and expiry date information.

See the VERSION log on *Page 233* for currently active models. Use the MODEL command, see *Page 76*, to change the currently active model. See the AUTH command on *Page 44* to add new models (up to a maximum of 5 models).

If a model has no expiry date it will report the year, month and day fields as 0, 0 and 0 respectively.

Structure: **Message ID: 206**
Log Type: Polled

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#mod	Number of models with information to follow.	Ulong	4	H
3	model	Model name	String [max. 16]	16	H+4
4	expyear	Expiry year	Ulong	4	H+20
5	expmonth	Expiry month	Ulong	4	H+24
6	expday	Expiry day	Ulong	4	H+28
7...	Next model offset = H + 4 + (#mods x 28)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4+ (#mods x 28)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log validmodelsa once

ASCII Example:

```
#VALIDMODELSA,COM1,0,78.5,FINESTEERING,1159,160078.661,00000000,0f9f,673;
2,"RT2WA",2003,12,31,"RT2",2003,12,31*94415c6d
```

3.4.82 VERSION Version Information

This log contains the version information for all components of a system. When using a standard receiver, there will only be one component in the log.

A component may be hardware (for example, a receiver or data collector) or firmware in the form of applications or data (for example, data blocks for height models, user applications or Field Programmable Gate-Array (FPGA) configurations). See *Table 78, Version Log: Field Formats on Page 234* for details on the format of key fields.

See also the VALIDMODELS log on *Page 232*.

Structure: **Message ID: 37**
Log Type: Polled

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	# comp	Number of components (cards, etc.)	Long	4	H
3	type	Component type (see *Table 77 on Page 234*)	Enum	4	H+4
4	model	Model	Char[16]	16	H+8
5	psn	Product serial number	Char[16]	16	H+24
6	hw version	Hardware version	Char[16]	16	H+40
7	sw version	Firmware software version	Char[16]	16	H+56
8	boot version	Boot code version	Char[16]	16	H+72
9	comp date	Firmware compile date	Char[12]	12	H+88
10	comp time	Firmware compile time	Char[12]	12	H+100
11...	Next component offset = H + 4 + (#comp x 108)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4+ (#comp x 108)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log versiona once

ASCII Example:

```
#VERSIONA,COM1,0,60.5,FINESTEERING,1039,493600.153,00000028,bc7a,0;
1,GPSCARD,"OEM4RT2","CPA99310032","OEM4-3.00A-22T","1.000S5","1.000",
"2000/Mar/17","14:48:55"*d5452b63
```

✉ The field types and formats remain constant for all components. However, unique applications' products may contain differences in the contents of the fields and their meanings. Below is an example of a VERSION log from a ProPak-LB receiver.

```
#VERSIONA,COM1,0,67.5,FINESTEERING,1163,485999.875,00000000,e249,710;4,
GPSCARD,"RT2WLBA","SPA02090052","OEM4-6.03-22T","1.400D86","1.005db","2002/
Apr/24","15:17:29",
DB_OMNISTARXILINX,"OmniXilinx","0","","102","","2002/Apr/11","10:56:48",
DB_OMNISTARDSP,"OmniDSP","0","","1.000S10","","2002/Apr/11","10:56:46",
IBOARD,"OMNISTAR","07f20040","6.01-102","1.000S10","704309","",""*74762c4f
```

Table 77: Component Types

Field	Format	Description
0	UNKNOWN	Unknown component
1	GPSCARD	OEM4 Family component
2	CONTROLLER	Data collector
4	IBOARD	OmniSTAR CAN[a] interface board
981073920	DB_HEIGHTMODEL	Height/track model data
981073921	DB_USERAPP	User application firmware
981073922	DB_OMNISTARDSP	OmniSTAR DSP[a] firmware
981073924	DB_OMNISTARX1LINX	OmniSTAR FPGA[a] firmware

a. Please refer to the Acronyms appendix in Volume 1 of this manual.

Table 78: Version Log: Field Formats

Component Type (binary)	Component Type (ASCII)	Description
hw version	P-RS-CCC	P = hardware platform (e.g. OEM4) R = hardware revision (e.g. 3.00) S = processor revision (e.g. A) CCC = COM port configuration (e.g. 22T)[a]
sw version, boot version	VV.RRR[Xxxx]	VV = major revision number RRR = minor revision number X = Special (S), Beta (B), Internal Development (D, A) xxx = number
comp date	YYYY/MM/DD	YYYY = year MM = month DD = day (1 - 31)
comp time	HH:MM:SS	HH = hour MM = minutes SS = seconds

a. One character for each of the COM ports 1, 2, and 3. Characters are: 2 for RS-232, 4 for RS-422 (available on Euro4 only) and T for LV-TTL.

3.4.83 WAAS0 Remove PRN from Solution SBAS

This message tells you, when you are using SBAS messages, not to use a specific PRN message for a period of time outlined in the SBAS signal specification.

See the WAASCORRECTION command on how the WAAS0 message relates to the SBAS testing modes.

Structure: **Message ID: 290**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Source PRN message - also PRN not to use.	Ulong	4	H
3	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4
4	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log WAAS0 onchanged

ASCII Example:

#WAAS0A,COM1,0,68.5,SATTIME,1093,161299.000,00040020,7d6a,209;122*e9a5ab08

3.4.84 WAAS1 PRN Mask Assignments SBAS

The PRN mask is given in WAAS1. The transition of the PRN mask to a new one (which will be infrequent) will be controlled with the 2-bit IODP, which will sequence to a number between 0 and 3. The same IODP will appear in the applicable WAAS2, WAAS3, WAAS4, WAAS5, WAAS7, WAAS24 and WAAS25 messages. This transition would probably only occur when a new satellite is launched or when a satellite fails and is taken out of service permanently. A degraded satellite may be flagged as a don't use satellite temporarily.

Structure: **Message ID: 291**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Source PRN of message.	Ulong	4	H
3	mask	PRN bit mask	Uchar[27]	28 [a]	H+4
4	iodp	Issue of PRN mask data.	Ulong	4	H+32
5	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+36
6	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 1 byte of padding is added to maintain 4 byte alignment

Recommended Input:

log WAAS1 onchanged

ASCII Example:

#WAAS1A,COM1,0,70.5,SATTIME,1093,161388.000,00040020,7e3f,209;122,
fffbfffe000000000000000000000040040000000000000000000,3*e4275412

3.4.85 WAAS2 Fast Correction Slots 0-12 SBAS

WAAS2 are fast corrections for slots 0-12 in the mask of WAAS1. This message may or may not come when SBAS is in testing mode (see the WAASCORRECTION command for details).

Structure: **Message ID: 296**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	
2	prn	Source PRN of message.	Ulong	4	H	-
4	iodf	Issue of fast corrections data.	Ulong	4	H+8	-
5	iodp	Issue of PRN mask data.	Ulong	4	H+12	-
6	prc0	prc(i): Fast corrections (-2048 to +2047) for the prn in slot i (i = 0-12).	Long	4	H+16	-
7	prc1		Long	4	H+20	-
8	prc2		Long	4	H+24	-
9	prc3		Long	4	H+28	-
10	prc4		Long	4	H+32	-
11	prc5		Long	4	H+36	-
12	prc6		Long	4	H+40	-
13	prc7		Long	4	H+44	-
14	prc8		Long	4	H+48	-
15	prc9		Long	4	H+52	-
16	prc10		Long	4	H+56	-
17	prc11		Long	4	H+60	-
18	prc12		Long	4	H+64	-
19	udre0	udre(i): User differential range error indicator for the prn in slot i (i = 0-12).	Ulong	4	H+68	See *Table 79*
20	udre1		Ulong	4	H+72	
21	udre2		Ulong	4	H+76	
22	udre3		Ulong	4	H+80	
23	udre4		Ulong	4	H+84	
24	udre5		Ulong	4	H+88	
25	udre6		Ulong	4	H+92	
26	udre7		Ulong	4	H+96	
27	udre8		Ulong	4	H+100	
28	udre9		Ulong	4	H+104	
29	udre10		Ulong	4	H+108	
30	udre11		Ulong	4	H+112	
31	udre12		Ulong	4	H+116	
32	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+120	-
33	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS2 onchanged

ASCII Example:

#WAAS2A,COM1,0,58.0,SATTIME,1093,163399.000,00000020,b4b0,209;
122,0,3,2047,3,-1,2047,2047,2047,-3,-1,5,3,3,2047,2,14,3,3,14,14,14,6,3,4,5,4,14,3*2e0894b1

Table 79: Evaluation of UDREI[1]

UDREI	UDRE meters	$\sigma^2_{i.udre}$ meters^2
0	0.75	0.0520
1	1.0	0.0924
2	1.25	0.1444
3	1.75	0.2830
4	2.25	0.4678
5	3.0	0.8315
6	3.75	1.2992
7	4.5	1.8709
8	5.25	2.5465
9	6.0	3.3260
10	7.5	5.1968
11	15.0	20.7870
12	50.0	230.9661
13	150.0	2078.695
14	Not Monitored	Not Monitored
15	Do Not Use	Do Not Use

1. The σ^2_{UDRE} broadcast in WAAS2, WAAS3, WAAS4, WAAS5, WAAS6 and WAAS24 applies at a time prior to or at the time of applicability of the associated corrections.

3.4.86 WAAS3 Fast Corrections Slots 13-25 SBAS

WAAS3 are fast corrections for slots 13-25 in the mask of WAAS1. This message may or may not come when SBAS is in testing mode (see the WAASCORRECTION command for details).

Structure: **Message ID: 301**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	
2	prn	Source PRN of message.	Ulong	4	H	-
4	iodf	Issue of fast corrections data.	Ulong	4	H+8	-
5	iodp	Issue of PRN mask data.	Ulong	4	H+12	-
6	prc13	prc(i): Fast corrections (-2048 to +2047) for the prn in slot i (i = 13-25).	Long	4	H+16	-
7	prc14		Long	4	H+20	-
8	prc15		Long	4	H+24	-
9	prc16		Long	4	H+28	-
10	prc17		Long	4	H+32	-
11	prc18		Long	4	H+36	-
12	prc19		Long	4	H+40	-
13	prc20		Long	4	H+44	-
14	prc21		Long	4	H+48	-
15	prc22		Long	4	H+52	-
16	prc23		Long	4	H+56	-
17	prc24		Long	4	H+60	-
18	prc25		Long	4	H+64	-
19	udrei13	udre(i): User differential range error indicator for the prn in slot i (i = 13-25).	Ulong	4	H+68	See *Table 79*
20	udrei14		Ulong	4	H+72	
21	udrei15		Ulong	4	H+76	
22	udrei16		Ulong	4	H+80	
23	udrei17		Ulong	4	H+84	
24	udrei18		Ulong	4	H+88	
25	udrei19		Ulong	4	H+92	
26	udrei20		Ulong	4	H+96	
27	udrei21		Ulong	4	H+100	
28	udrei22		Ulong	4	H+104	
29	udrei23		Ulong	4	H+108	
30	udrei24		Ulong	4	H+112	
31	udrei25		Ulong	4	H+116	
32	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+120	-
33	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS3 onchanged

ASCII Example:

#WAAS3A,COM1,0,72.5,SATTIME,1093,161480.000,00040020,31d4,209;122,1,3,-7,2047,2047,2047,
-4,2047,2047,2047,9,2047,2047,-3,-2,11,14,14,14,4,14,14,14,5,14,14,4,2*2bf0109b

3.4.87 WAAS4 Fast Correction Slots 26-38 SBAS

WAAS4 are fast corrections for slots 26-38 in the mask of WAAS1. This message may or may not come when SBAS is in testing mode (see the WAASCORRECTION command for details).

Structure: **Message ID: 302**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	
2	prn	Source PRN of message.	Ulong	4	H	-
4	iodf	Issue of fast corrections data.	Ulong	4	H+8	-
5	iodp	Issue of PRN mask data.	Ulong	4	H+12	-
6	prc26	prc(i): Fast corrections (-2048 to +2047) for the prn in slot i (i = 26-38).	Long	4	H+16	-
7	prc27		Long	4	H+20	-
8	prc28		Long	4	H+24	-
9	prc29		Long	4	H+28	-
10	prc30		Long	4	H+32	-
11	prc31		Long	4	H+36	-
12	prc32		Long	4	H+40	-
13	prc33		Long	4	H+44	-
14	prc34		Long	4	H+48	-
15	prc35		Long	4	H+52	-
16	prc36		Long	4	H+56	-
17	prc37		Long	4	H+60	-
18	prc38		Long	4	H+64	-
19	udrei26	udre(i): User differential range error indicator for the prn in slot i (i = 26-38).	Ulong	4	H+68	See *Table 79*
20	udrei27		Ulong	4	H+72	
21	udrei28		Ulong	4	H+76	
22	udrei29		Ulong	4	H+80	
23	udrei30		Ulong	4	H+84	
24	udrei31		Ulong	4	H+88	
25	udrei32		Ulong	4	H+92	
26	udrei33		Ulong	4	H+96	
27	udrei34		Ulong	4	H+100	
28	udrei35		Ulong	4	H+104	
29	udrei36		Ulong	4	H+108	
30	udrei37		Ulong	4	H+112	
31	udrei38		Ulong	4	H+116	
32	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+120	-
33	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS4 onchanged

ASCII Example:

#WAAS4A,COM1,0,58.0,SATTIME,1093,163399.000,00000020,b4b0,209;
122,0,3,2047,3,-1,2047,2047,2047,-3,-1,5,3,3,2047,2,14,3,3,14,14,14,6,3,4,5,4,14,3*2e0894b1

3.4.88 WAAS5 Fast Correction Slots 39-50 SBAS

WAAS5 are fast corrections for slots 39-50 in the mask of WAAS1. This message may or may not come when SBAS is in testing mode (see the WAASCORRECTION command for details).

Structure: **Message ID: 303**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	
2	prn	Source PRN of message.	Ulong	4	H	-
4	iodf	Issue of fast corrections data.	Ulong	4	H+8	-
5	iodp	Issue of PRN mask data.	Ulong	4	H+12	-
6	prc39	prc(i): Fast corrections (-2048 to +2047) for the prn in slot i (i = 39-50).	Long	4	H+16	-
7	prc40		Long	4	H+20	-
8	prc41		Long	4	H+24	-
9	prc42		Long	4	H+28	-
10	prc43		Long	4	H+32	-
11	prc44		Long	4	H+36	-
12	prc45		Long	4	H+40	-
13	prc46		Long	4	H+44	-
14	prc47		Long	4	H+48	-
15	prc48		Long	4	H+52	-
16	prc49		Long	4	H+56	-
17	prc50		Long	4	H+60	-
18	prc51 (Invalid, do not use)		Long	4	H+64	-
19	udrei39	udre(i): User differential range error indicator for the prn in slot i (i = 39-50).	Ulong	4	H+68	See *Table 79*
20	udrei40		Ulong	4	H+72	
21	udrei41		Ulong	4	H+76	
22	udrei42		Ulong	4	H+80	
23	udrei43		Ulong	4	H+84	
24	udrei44		Ulong	4	H+88	
25	udrei45		Ulong	4	H+92	
26	udrei46		Ulong	4	H+96	
27	udrei47		Ulong	4	H+100	
28	udrei48		Ulong	4	H+104	
29	udrei49		Ulong	4	H+108	
30	udrei50		Ulong	4	H+112	
31	udrei51 (Invalid, do not use)		Ulong	4	H+116	
32	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+120	-
33	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS5 onchanged

ASCII Example:

#WAAS5A,COM1,0,72.5,SATTIME,1093,161480.000,00040020,31d4,209;122,1,3,-7,2047,2047,2047,
-4,2047,2047,2047,9,2047,2047,-3,-2,11,14,14,14,4,14,14,14,5,14,14,4,2*2bf0109b

3.4.89 WAAS6 Integrity Message SBAS

WAAS6 is the integrity information message. Each message includes an IODF for each fast corrections message. The s^2_{UDRE} information for each block of satellites applies to the fast corrections with the corresponding IODF.

Structure: **Message ID: 304**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	iodf2	Issue of fast corrections data.	Ulong	4	H+4	-
4	iodf3	Issue of fast corrections data.	Ulong	4	H+8	-
5	iodf4	Issue of fast corrections data.	Ulong	4	H+12	-
6	iodf5	Issue of fast corrections data.	Ulong	4	H+16	-
7	udrei0	udre(i): User differential range error indicator for the prn in slot i (i = 0-50).	Ulong	4	H+20	See *Table 79* on *Page 238*
8	udrei1		Ulong	4	H+24	
9	udrei2		Ulong	4	H+28	
10	udrei3		Ulong	4	H+32	
11	udrei4		Ulong	4	H+36	
12	udrei5		Ulong	4	H+40	
13	udrei6		Ulong	4	H+44	
14	udrei7		Ulong	4	H+48	
15	udrei8		Ulong	4	H+52	
16	udrei9		Ulong	4	H+56	
17	udrei10		Ulong	4	H+60	
18	udrei11		Ulong	4	H+64	
19	udrei12		Ulong	4	H+68	
20	udrei13		Ulong	4	H+72	
21	udrei14		Ulong	4	H+76	
22	udrei15		Ulong	4	H+80	
23	udrei16		Ulong	4	H+84	
24	udrei17		Ulong	4	H+88	
25	udrei18		Ulong	4	H+92	
26	udrei19		Ulong	4	H+96	
27	udrei20		Ulong	4	H+100	
28	udrei21		Ulong	4	H+104	
29	udrei22		Ulong	4	H+108	
30	udrei23		Ulong	4	H+112	
31	udrei24		Ulong	4	H+116	
32	udrei25		Ulong	4	H+120	
33	udrei26		Ulong	4	H+124	

34	udrei27		Ulong	4	H+128	
35	udrei28		Ulong	4	H+132	
36	udrei29		Ulong	4	H+136	
37	udrei30		Ulong	4	H+140	
38	udrei31		Ulong	4	H+144	
39	udrei32		Ulong	4	H+148	
40	udrei33		Ulong	4	H+152	
41	udrei34		Ulong	4	H+156	
42	udrei35		Ulong	4	H+160	
43	udrei36		Ulong	4	H+164	
44	udrei37		Ulong	4	H+168	
45	udrei38		Ulong	4	H+172	
46	udrei39		Ulong	4	H+176	
47	udrei40		Ulong	4	H+180	
48	udrei41		Ulong	4	H+184	
49	udrei42		Ulong	4	H+188	
50	udrei43		Ulong	4	H+192	
51	udrei44		Ulong	4	H+196	
52	udrei45		Ulong	4	H+200	
53	udrei46		Ulong	4	H+204	
54	udrei47		Ulong	4	H+208	
55	udrei48		Ulong	4	H+212	
56	udrei49		Ulong	4	H+216	
58	udrei50		Ulong	4	H+220	
58	udrei51 (Invalid, do not use)		Ulong	4	H+224	
59	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+228	-
60	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS6 onchanged

ASCII Example:

#WAAS6A,COM1,0,57.5,SATTIME,1093,273317.000,00000020,526a,209;
122,3,3,3,3,9,14,14,2,3,10,2,14,14,3,14,14,5,14,14,7,14,14,14,14,14,14,3,3,14,14,14,14,3,15,11,11,15,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0*925a2a9b

3.4.90 WAAS7 Fast Correction Degradation SBAS

The WAAS7 message specifies the applicable IODP, system latency time and fast degradation factor indicator for computing the degradation of fast and long-term corrections.

Structure: **Message ID: 305**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Source PRN of message.	Ulong	4	H
3	latency	System latency.	Ulong	4	H+4
4	iodp	Issue of PRN mask data.	Ulong	4	H+8
5	spare bits	Unused spare bits.	Ulong	4	H+12
6	aI(0)	aI(i): Degradation factor indicator for the prn in slot i (i = 0-50).	Ulong	4	H+16
7	aI(1)		Ulong	4	H+20
8	aI(2)		Ulong	4	H+24
9	aI(3)		Ulong	4	H+28
10	aI(4)		Ulong	4	H+32
11	aI(5)		Ulong	4	H+36
12	aI(6)		Ulong	4	H+40
13	aI(7)		Ulong	4	H+44
14	aI(8)		Ulong	4	H+48
15	aI(9)		Ulong	4	H+52
16	aI(10)		Ulong	4	H+56
17	aI(11)		Ulong	4	H+60
18	aI(12)		Ulong	4	H+64
19	aI(13)		Ulong	4	H+68
20	aI(14)		Ulong	4	H+72
21	aI(15)		Ulong	4	H+76
22	aI(16)		Ulong	4	H+80
23	aI(17)		Ulong	4	H+84
24	aI(18)		Ulong	4	H+88
25	aI(19)		Ulong	4	H+92
26	aI(20)		Ulong	4	H+96
27	aI(21)		Ulong	4	H+100
28	aI(22)		Ulong	4	H+104
29	aI(23)		Ulong	4	H+108
30	aI(24)		Ulong	4	H+112
31	aI(25)		Ulong	4	H+116
32	aI(26)		Ulong	4	H+120
33	aI(27)		Ulong	4	H+124
34	aI(28)		Ulong	4	H+128

35	aI(29)		Ulong	4	H+132
36	aI(30)		Ulong	4	H+136
37	aI(31)		Ulong	4	H+140
38	aI(32)		Ulong	4	H+144
39	aI(33)		Ulong	4	H+148
40	aI(34)		Ulong	4	H+152
41	aI(35)		Ulong	4	H+156
42	aI(36)		Ulong	4	H+160
43	aI(37)		Ulong	4	H+164
44	aI(38)		Ulong	4	H+168
45	aI(39)		Ulong	4	H+172
46	aI(40)		Ulong	4	H+176
47	aI(41)		Ulong	4	H+180
48	aI(42)		Ulong	4	H+184
49	aI(43)		Ulong	4	H+188
50	aI(44)		Ulong	4	H+192
51	aI(45)		Ulong	4	H+196
52	aI(46)		Ulong	4	H+200
53	aI(47)		Ulong	4	H+204
54	aI(48)		Ulong	4	H+208
55	aI(49)		Ulong	4	H+212
56	aI(50)		Ulong	4	H+216
57	aI(51) (Invalid, do not use)		Ulong	4	H+220
58	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+224
59	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log WAAS7 onchanged

ASCII Example:

#WAAS7A,COM1,0,62.0,SATTIME,1093,161555.000,00040020,1706,209;122,1,3,0,15,15,15,15,15,15,
15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,15,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0*902e0aea

3.4.91 WAAS9 GEO Navigation Message SBAS

WAAS9 provides the GEO navigation message representing the position, velocity and acceleration of the geostationary satellite, in ECEF coordinates and its apparent clock time and frequency offsets.

Also included is the time of applicability, an issue of data (IOD) and an accuracy exponent (URA) representing the estimated accuracy of the message. The time offset and time drift are with respect to SBAS Network Time. Their combined effect is added to the estimate of the satellite's transmit time.

Structure: **Message ID: 306**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Source PRN of message.	Ulong	4	H
3	iodn	Issue of GEO navigation data.	Ulong	4	H+4
4	t_0	Time of applicability	Ulong	4	H+8
5	ura	URA value	Ulong	4	H+12
6	x	ECEF x coordinate	Double	8	H+16
7	y	ECEF y coordinate	Double	8	H+24
8	z	ECEF z coordinate	Double	8	H+32
9	xvel	X rate of change	Double	8	H+40
10	yvel	Y rate of change	Double	8	H+48
11	zvel	Z rate of change	Double	8	H+56
12	xaccel	X rate of rate change	Double	8	H+64
13	yaccel	Y rate of rate change	Double	8	H+72
14	zaccel	Z rate of rate change	Double	8	H+80
15	a_{f0}	Time offset	Double	8	H+88
16	a_{f1}	Time drift	Double	8	H+96
17	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+104
18	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log WAAS9 onchanged

ASCII Example:

#WAAS9A,COM1,0,69.0,SATTIME,1093,161622.000,00040020,8a3d,209;
122,72,75072,0,24792366.0000,-34095207.2800,38650.8000,14.931250000,-0.168750000,4.0800000,
-0.0000750,-0.0001125,-0.000187500,5.587935448e-09,7.275957614e-12*18bc92b6

3.4.92 WAAS10 Degradation Factor SBAS

The fast corrections, long-term corrections and ionospheric corrections are all provided in the WAAS10 message.

Structure: **Message ID: 292**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	b_{rcc}	Estimated noise and round off error parameter.	Ulong	4	H+4	0.002
4	c_{ltc_lsb}	Maximum round off due to the lest significant bit (lsb) of the orbital clock.	Ulong	4	H+8	0.002
5	c_{ltc_vl}	Velocity error bound.	Ulong	4	H+12	0.00005
6	i_{ltc_vl}	Update interval for v=1 long term.	Ulong	4	H+16	-
7	c_{ltc_v0}	Bound on update delta.	Ulong	4	H+20	0.002
8	i_{ltc_v1}	Minimum update interval v = 0.	Ulong	4	H+24	-
9	c_{geo_lsb}	Maximum round off due to the lsb of the orbital clock.	Ulong	4	H+28	0.0005
10	c_{geo_v}	Velocity error bound.	Ulong	4	H+32	0.00005
11	i_{geo}	Update interval for GEO navigation message.	Ulong	4	H+36	-
12	c_{er}	Degradation parameter.	Ulong	4	H+40	0.5
13	c_{iono_step}	Bound on ionospheric grid delay difference.	Ulong	4	H+44	0.001
14	i_{iono}	Minimum ionospheric update interval.	Ulong	4	H+48	-
15	c_{iono_ramp}	Rate of ionospheric corrections change.	Ulong	4	H+52	0.000005
16	rss_{udre}	User differential range error flag.	Ulong	4	H+56	-
17	rss_{iono}	Root sum square flag.	Ulong	4	H+60	-
18	spare bits	Spare 88 bits, possibly GLONASS.	Ulong	4	H+64	-
19	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+68	-
20	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS10 onchanged

ASCII Example:

#WAAS10A,COM1,0,61.0,SATTIME,1093,161669.000,00040020,e520,209;122,54,38,76,256,152,30,311,83,256,6,122,270,0,0,0,00000000000000000000000*aee479f8

3.4.93 WAAS12 SBAS Network Time and UTC *SBAS*

WAAS12 consists of the 8-bit preamble, a 6-bit message type identifier (= 12) followed by 104 information bits for the UTC parameters, then followed by 3 bits to indicate the UTC time standard from which the offset is determined.

The UTC parameters will correlate UTC time with the SBAS network time rather than with GPS time.

Structure: **Message ID: 293**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Source PRN of message.	Ulong	4	H
3	A_1	Time offset.	Double	8	H+4
4	A_0	Time drift.	Double	8	H+12
5	seconds	Seconds into the week.	Ulong	4	H+20
6	week	Week number.	Ushort	4	H+24
7	dt_{ls}	Delta time due to leap seconds.	Short	2	H+28
8	wn_{lsf}	Week number, leap second future.	Ushort	2	H+30
9	dn	Day of the week.	Ushort	2	H+32
10	dt_{lsf}	Delta time, leap second future.	Short	2	H+34
11	utc id	UTC type identifier.	Ushort	2	H+36
12	gpstow	GPS time of the week.	Ulong	2	H+38
13	gpswn	GPS de-modulo week number.	Ulong	2	H+40
14	glo indicator	Is GLONASS information present? 0 = FALSE 1 = TRUE	Enum	4	H+42
15	Reserved		Char[10]	12[a]	H+46
16	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+58
17	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log WAAS12 onchanged

ASCII Example:

Not available at time of print.

3.4.94 WAAS17 GEO Almanac Message SBAS

Almanacs for all GEOs will be broadcast periodically to alert you of their existence, location, the general service provides, status, and health.

Structure: **Message ID: 294**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	#ents	Number of almanac entries with information to follow.	Ulong	4	H+4	-
4	data id	Data ID type.	Ushort	2	H+8	-
5	entry prn	PRN for this entry.	Ushort	2	H+10	-
6	health	Health bits.	Ushort	4[a]	H+12	-
7	x	ECEF x coordinate.	Long	4	H+16	-
8	y	ECEF y coordinate.	Long	4	H+20	-
9	z	ECEF z coordinate.	Long	4	H+24	-
10	x vel	X rate of change.	Long	4	H+28	-
11	y vel	Y rate of change.	Long	4	H+32	-
12	z vel	Z rate of change.	Long	4	H+36	-
13...	Next entry = H+8 + (#ents x 32)					-
variable	t0	Time of day in seconds (0 to 86336)	Ulong	4	H+8+ (#ents x 32)	64
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+12+ (#ents x 32)	-
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

a. In the binary log case an additional 2 bytes of padding is added to maintain 4 byte alignment

Recommended Input:

log WAAS17 onchanged

ASCII Example:

```
#WAAS17A,COM1,0,61.0,SATTIME,1093,161820.000,00040020,5116,209;
122,3,0,134,0,-42127800,1518400,-52000,0,0,0,0,122,0,24793600,-34096400,52000,
0,0,0,0,0,0,0,0,0,0,0,0*c040d1f4
```

3.4.95 WAAS18 IGP Mask SBAS

The ionospheric delay corrections are broadcast as vertical delay estimates at specified ionospheric grid points (IGPs), applicable to a signal on L1. The predefined IGPs are contained in 11 bands (numbered 0 to 10). Bands 0-8 are vertical bands on a Mercator projection map, and bands 9-10 are horizontal bands on a Mercator projection map. Since it is impossible to broadcast IGP delays for all possible locations, a mask is broadcast to define the IGP locations providing the most efficient model of the ionosphere at the time.

Structure: **Message ID: 295**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	prn	Source PRN of message.	Ulong	4	H
3	#bands	Number of bands broadcast.	Ulong	4	H+4
4	band num	Specific brand number.	Ulong	4	H+8
5	iodi	Issue of ionospheric data.	Ulong	4	H+12
6	igp mask	IGP mask.	Uchar[26]	28[a]	H+16
7	spare bit	One spare bit.	Ulong	4	H+44
8	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+48
9	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

a. In the binary log case an additional 2 bytes of padding are added to maintain 4 byte alignment

Recommended Input:

log WAAS18 onchanged

ASCII Example:

#WAAS18A,COM1,0,68.0,SATTIME,1093,161963.000,00000020,f88e,209;
122,4,3,0,0001ff000000000000000000000000000000000000000000000000,0*b03300f3

3.4.96 WAAS24 Mixed Fast/Slow Corrections SBAS

If there are 6 or fewer satellites in a block, they may be placed in this mixed correction message.The fast data set for each satellite consists of 16 bits; a 12-bit fast correction and a 4-bit UDRE indicator. Each message also contains a 2-bit IODP indicating the associated PRN mask.

The 12-bit fast correction (PRC) has a 0.125 meter resolution, for a valid range of -256 to +255.875 m. If the range is exceeded a don't use indication will be inserted into the UDREI field. You should ignore extra data sets not represented in the PRN mask.

The time of applicability (T0) of the PRC is the start of the epoch of the WNT second that is coincident with the transmission at the GEO satellite of the first bit of the message block.

Structure: **Message ID: 297**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	prc0	prc(i): Fast corrections (-2048 to +2047) for the prn in slot i (i = 0-5).	Long	4	H+4	-
4	prc1		Long	4	H+8	-
5	prc2		Long	4	H+12	-
6	prc3		Long	4	H+16	-
7	prc4		Long	4	H+20	-
8	prc5		Long	4	H+24	-
9	udrei0	udre(i): User differential range error indicator for the prn in slot i (i = 0-5).	Ulong	4	H+28	See *Table 79* on *Page 238*
10	udrei1		Ulong	4	H+.32	
11	udrei2		Ulong	4	H+36	
12	udrei3		Ulong	4	H+40	
13	udrei4		Ulong	4	H+44	
14	udrei5		Ulong	4	H+48	
15	iodp	Issue of PRN mask data.	Ulong	4	H+52	-
16	block id	Associated message type.	Ulong	4	H+56	
17	iodf	Issue of fast corrections data.	Ulong	4	H+60	-
18	spare	Spare value.	Ulong	4	H+64	-
19	vel	Velocity code flag.	Ulong	4	H+68	-
20	mask1	Index into PRN mask (Type 1).	Ulong	4	H+72	-
21	iode1	Issue of ephemeris data.	Ulong	4	H+76	-
22	dx1	Delta x (ECEF).	Long	4	H+80	0.125
23	dy1	Delta y (ECEF).	Long	4	H+84	0.125
24	dz1	Delta z (ECEF).	Long	4	H+88	0.125
25	da^{f0}	Delta a^{f0} clock offset.	Long	4	H+92	2^{-31}
26	mask2	Second index into PRN mask (Type 1).	Ulong	4	H+96	-
27	iode2	Second issue of ephemeris data.	Ulong	4	H+100	-
28	ddx	Delta delta x (ECEF).	Long	4	H+104	2^{-11}
29	ddy	Delta delta y (ECEF).	Long	4	H+108	2^{-11}

30	ddz	Delta delta z (ECEF).	Long	4	H+112	2^{-11}
31	da^{f1}	Delta a^{f1} clock offset.	Long	4	H+116	2^{-39}
32	t_0	Applicable time of day.	Ulong	4	H+120	16
33	iodp	Issue of PRN mask data.	Ulong	4	H+124	-
34	corr spare	Spare value when velocity code = 0.	Ulong	4	H+128	-
35	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+132	-
36	[CR][LF]	Sentence terminator (ASCII only)	-	-	H+136	-

Recommended Input:

log WAAS24 onchanged

ASCII Example:

#WAAS24A,COM1,0,63.5,SATTIME,1093,162057.000,00000020,c0f9,209;
122,1,2047,2047,0,-11,220,1,14,14,3,11,11,3,2,2,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0*cea865a5

3.4.97 WAAS25 Long-Term Slow Satellite Corrections SBAS

WAAS25 provides error estimates for slow varying satellite ephemeris and clock errors with respect to WGS-84 ECEF coordinates.

Structure: **Message ID: 298**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	#1st	Number first half with information to follow.	Ulong	4	H+4	-
4	vel	Velocity code flag.	Ulong	4	H+8	-
5	mask1	Index into PRN mask (Type 1).	Ulong	4	H+12	-
6	iode1	Issue of ephemeris data.	Ulong	4	H+16	-
7	dx1	Delta x (ECEF).	Long	4	H+20	0.125
8	dy1	Delta y (ECEF).	Long	4	H+24	0.125
9	dz1	Delta z (ECEF).	Long	4	H+28	0.125
10	a^{f0}	Delta a^{f0} clock offset.	Long	4	H+32	2^{-31}
11	mask2	Second index into PRN mask (Type 1).	Ulong	4	H+36	-
12	iode2	Second issue of ephemeris data.	Ulong	4	H+40	-
13	ddx	Delta delta x (ECEF).	Long	4	H+44	2^{-11}
14	ddy	Delta delta y (ECEF).	Long	4	H+48	2^{-11}
15	ddz	Delta delta z (ECEF).	Long	4	H+52	2^{-11}
16	a^{f1}	Delta a^{f1} clock offset.	Long	4	H+56	2^{-39}
17	t_0	Applicable time of day.	Ulong	4	H+60	16
18	iodp	Issue of PRN mask data.	Ulong	4	H+64	-
19	corr spare	Spare value when velocity code = 0.	Ulong	4	H+68	-
20...	Next #1st = H +8 + (#1st x 64)					
variable	#2nd	Number of second half with information to follow.	Ulong	4	H+8+ (#1st x 64)	
variable	vel	Velocity code flag.	Ulong	4	H+12+ (#1st x 64)	-
variable	mask1	Index into PRN mask (Type 1).	Ulong	4	H+16+ (#1st x 64)	-
variable	iode1	Issue of ephemeris data.	Ulong	4	H+20+ (#1st x 64)	-
variable	dx1	Delta x (ECEF).	Long	4	H+24+ (#1st x 64)	0.125
variable	dy1	Delta y (ECEF).	Long	4	H+28+ (#1st x 64)	0.125
variable	dz1	Delta z (ECEF).	Long	4	H+32+ (#1st x 64)	0.125

variable	a^{f0}	Delta a^{f0} clock offset.	Long	4	H+36+ (#1st x 64)	2^{-31}
variable	mask2	Second index into PRN mask (Type 1).	Ulong	4	H+40+ (#1st x 64)	-
variable	iode2	Second issue of ephemeris data.	Ulong	4	H+44+ (#1st x 64)	-
variable	ddx	Delta delta x (ECEF).	Long	4	H+48+ (#1st x 64)	2^{-11}
variable	ddy	Delta delta y (ECEF).	Long	4	H+52+ (#1st x 64)	2^{-11}
variable	ddz	Delta delta z (ECEF).	Long	4	H+56+ (#1st x 64)	2^{-11}
variable	a^{f1}	Delta a^{f1} clock offset.	Long	4	H+60+ (#1st x 64)	2^{-39}
variable	t_0	Applicable time of day.	Ulong	4	H+64+ (#1st x 64)	16
variable	iodp	Issue of PRN mask data.	Ulong	4	H+68+ (#1st x 64)	-
variable	corr spare	Spare value when velocity code = 0.	Ulong	4	H+72+ (#1st x 64)	-
variable	Next #2nd = variable					
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	variable	-
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS25 onchanged

ASCII Example:

#WAAS25A,COM1,0,69.0,SATTIME,1093,162120.000,00000020,b2c2,209;
122,1,14,182,-66,146,-48,-12,0,0,-2,2,-6,-2,4723,3,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0*4dbed001

3.4.98 WAAS26 Ionospheric Delay Corrections SBAS

WAAS26 provides vertical delays (relative to an L1 signal) and their accuracy at geographically defined IGPs identified by the BAND NUMBER and IGP number. Each message contains a band number and a block ID, which indicates the location of the IGPs in the respective band mask.

Structure: **Message ID: 299**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	band num	Band number.	Ulong	4	H+4	-
4	block id	Block ID.	Ulong	4	H+8	-
5	#pts	Number of grid points with information to follow.	Ulong	4	H+12	-
6	igp_{vde}	IGP vertical delay estimates	Ulong	4	H+16	0.125
7	givei	Grid ionospheric vertical error indicator.	Ulong	4	H+20	-
8...	Next #pts entry = H + 16 + (#pts x 8)					
variable	iodi	Issue of data - ionosphere	Ulong	4	H+16+ (#pts x 8)	
variable	spare	7 spare bits.	Ulong	4	H+20+ (#pts x 8)	-
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+24+ (#pts x 8)	-
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS26 onchanged

ASCII Example:

#WAAS26A,COM1,0,68.5,SATTIME,1122,333809.000,00000000,a76c,459;
122,2,1,15,26,15,50,15,42,15,32,3,27,2,24,1,22,2,23,15,28,15,57,15,46,15,39,4,33,3,27,3,24,2,0,
0*166bc58a

3.4.99 WAAS27 SBAS Service Message *SBAS*

WAAS27 messages apply only to the service provider transmitting the message. The number of service messages indicates the total number of unique WAAS27 messages for the current IODS. Each unique message for that IODS includes a sequential message number. The IODS is incremented in all messages, each time that any parameter in any WAAS27 message is changed.

Structure: **Message ID: 300**
Log Type: Asynch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset	Scaling
1	header	Log header		H	0	-
2	prn	Source PRN of message.	Ulong	4	H	-
3	iods	Issue of slow corrections data.	Ulong	4	H+4	-
4	#messages	Low-by-one count of messages.	Ulong	4	H+8	-
5	message num	Low-by-one message number.	Ulong	4	H+12	-
6	priority code	Priority code.	Ulong	4	H+16	-
7	dudre inside	Delta user differential range error - inside.	Ulong	4	H+20	-
8	dudre outside	Delta user differential range error - outside.	Ulong	4	H+24	-
9...	#reg	Number of regions with information to follow.	Ulong	4	H+28	-
variable	lat1	Coordinate 1 latitude.	Long	4	H+32	-
variable	lon1	Coordinate 1 longitude.	Long	4	H+36	-
variable	lat2	Coordinate 2 latitude.	Long	4	H+40	-
variable	lon2	Coordinate 2 longitude.	Long	4	H+44	-
variable	shape	Shape where 0 = triangle and 1 = square.	Ulong	4	H+48	-
variable	Next #reg entry = H + 32 + (#reg x 20)					
variable	t_0	Time of applicability	Ulong	4	H+32+ (#reg x 20)	16
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+36+ (#reg x 20)	-
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-	-

Recommended Input:

log WAAS27 onchanged

ASCII Example:

Not available at time of print.

3.4.100 WAASCORR SBAS Range Corrections Used SBAS

The info will be updated with each pseudorange position calculation. It will have an entry for each tracked satellite. Satellites that are not included in an SBAS corrected solution will have 0.0 in both the ‘psr corr’ and ‘corr stdv’ fields.

The ‘psr corr’ is the combined fast and slow corrections and is to be added to the pseudorange. Ionospheric and tropospheric corrections are not included and should be applied separately.

Structure: **Message ID: 313**
Log Type: Synch

Field #	Field type	Data Description	Format	Binary Bytes	Binary Offset
1	header	Log header		H	0
2	#sat	Number of satellites with information to follow.	Ulong	4	H
3	prn	Satellite PRN.	Ulong	4	H+4
4	iode	Issue of ephemeris data for which the corrections apply.	Ulong	4	H+8
5	psr corr	SBAS pseudorange correction. (m)	Float	4	H+12
6	corr stdv	Standard deviation of pseudorange correction. (m)	Float	4	H+16
7...	Next sat entry = H+4 + (#sat x 16)				
variable	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+4+ (#sat x 16)
variable	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

Recommended Input:

log waascorr ontime 1

ASCII Example:

#WAASCORRA,COM1,0,65.5,FINESTEERING,1093,164788.000,00000020,a71d,209;
9,23,23,2.9357,1.7761,2,16,0.1643,1.7964,27,78,2.1286,1.2761,11,10,1.1836,1.7964,7,171,-1.0310,
3.0464,8,200,1.3313,1.2964,26,50,2.2381,3.7761,31,135,1.8975,1.4356,122,0,0.0000,0.0000*abcc3573

Chapter 4 Responses

The receiver is capable of outputting several responses for various conditions. Most of these responses are error messages to indicate when something is not correct.

The output format of the messages is dependant on the format of the input command. If the command is input as abbreviated ASCII, the output will be abbreviated ASCII. Likewise for ASCII and binary formats. *Table* outlines the various responses.

Table 80: Response Messages

ASCII Message	Binary Message ID	Meaning
Response_OK	1	Command was received correctly.
LogDoesNotExist	2	The log requested does not exist.
InsufficientResources	3	The request has exceeded a limit (e.g. the maximum number of logs are being generated).
Response_Verify_Fail	4	Data packet is not verified
Response_Fail	5	Command did not succeed in accomplishing requested task.
Invalid_Message_ID	6	The input message ID is not valid.
Invalid_Message_Format	7	Field *x* of the input message is not correct.
Invalid_Checksum	8	The checksum of the input message is not correct. This only applies to ASCII and binary format messages.
Response_Missing_Required_Field	9	Field *x* is missing from the input message.
Response_Index_Out_Of_Range	10	Field *x* contains more array elements than allowed.
Response_Parameter_Out_Of_Range	11	Field *x* of the input message is outside the acceptable limits.
Invalid_Trigger	14	Input trigger (ONTIME, etc.) is not valid for this type of log.
Auth_Table_Full	15	Too many authcodes are stored in the receiver. The receiver firmware must be reloaded.
Auth_Invalid_Date	16	This error is related to the inputting of authcodes. It indicates that the date attached to the code is not valid.
Auth_Invalid Parameter	17	The authcode entered is not valid.
Auth_No_Match	18	The model requested for removal does not exist.
No_Model	19	The model attached to the authcode is not valid.
Invalid_Channel	20	The selected log cannot be output at the specified rate.
Response_Invalid_Rate	21	The requested rate is invalid.
RxStatus_NoMask	22	Word y has no mask for this type of log.
Channel_Lockout	23	Channels are locked due to error.
Invalid_Inject_Time	24	Injected time is invalid
Invalid_COM_Port	25	The COM port is not supported.
Invalid_Binary_Msg	26	The message is invalid.
Invalid_PRN	27	The PRN is invalid.
PRN_Not_Locked_Out	28	The PRN is not locked out.

Table 80: Response Messages

PRN_Lockout_List_Full	29	PRN lockout list is full.
PRN_Already_Locked_Out	30	The PRN is already locked out.
Response_Time_Out	31	Message timed out.
UnknownPort	33	Unknown COM port requested.
InvalidHexString	34	Hex string not formatted correctly.
Invalid_Baud_Rate	35	The baud rate is invalid.
Response_Parameter_Msg_Invalid	36	This message is invalid for this model of receiver.
NVM_Not_Failed	40	Command is only valid if NVM is in fail mode
Response_Invalid_Offset	41	The offset is invalid.
NVM_Msgdef_Full	78	Maximum number of user messages has been reached.
Precise_Time_Known	84	GPS precise time is already known.

Index

Numerics

1PPS, *see* one pulse per second
2-D 66, 145
3-D 66, 145, 172

A

abbreviated ascii 12, 15
accumulated Doppler range (ADR) 177
accuracy
  corrections 65
  degradation 122
  input position 82
  limits 78
  position 65, 83
  RTK solution 106
  time 19
acquisition 39, 65–66, 98, 179
ADJUST1PPS 35
ADR, *see* accumulated Doppler range
age
  differential
    position 122, 157, 165, 173, 214
    RTK 106–107, 214–215
    velocity 125–126, 174, 176, 217
    xyz cartesian coordinate system 126, 176, 217
  solution 122, 126, 157, 165, 173, 176, 214, 217
ALMANAC 119
almanac
  complete 226
  data 140
  log 108–109, 119
  lost 77
  raw data 182
  reset 68
  stored 68
  time status 19
along track 162–163
ambiguity
  half cycle 177
  resolution speed 82
  type 87, 212–213
anomaly 119, 231
antenna
  active 38
  altitude 61, 141–142
  baseline 82
  motion 40–41, 85, 175
  phase center 66
  position 157
  power 97
  receiver status 220
  speed 163
ANTENNAPOWER 38
anti-spoofing (AS) 119
AS, *see* anti-spoofing
ascii
  message 13, 23
  overview 13
  printable data 91
  redirect 170
  response 18
  send 91
  text message 96
ASSIGN 39
assign
  all channels 41
  cancel 98
  channel 24, 39
  cut-off angle 61
ASSIGNALL 41
ASSIGNOMNI 43
asterisk 13
asynchronous logs 104
atmospheric
  delays 177
  refraction 61
AUTH 44
authorization 23, 44–45
auxiliary status 224
AVEPOS 121
averaging, position 25, 78, 121
azimuth 147, 226

B

bandwidth 89
base station
  commands 25
  logs 108
  type 189
baseline
  command 82–83
  float solution 212
  length 86
  RTK 24, 108
  static 26
baud rate, *see* bps
bearing 74–75, 148, 162–163
BESTPOS 122
BESTVEL 125
BESTXYZ 126
bias 46, 107
bi-directional communication 170
binary
  header 16
  overview 15
  raw ephemeris 183
  redirect 170
  response 18
bit rate, *see* bps
bps 49, 77, 171
break 49, 171
broadcast almanac 140
buffer 71, 213
byte 13, 16

C

C/N0, *see* carrier to noise density ratio
carrier phase 46, 132, 175, 177–178, 193
carrier to noise density ratio (C/N0) 147, 168, 178, 181, 230
caution 90
channel
  assign 39–40
  control 24, 109
  range measurements 177
  raw subframe data 184, 188
  tracking status 177, 181, 230
  unassign 98
checksum 13, 15
clock
  adjustment 46, 177
  aging parameter 119
  bias 46
  commands 26
  dither 62, 128
  drift 46
  error 40–41, 46, 128
  internal 19
  model 128–129
  offset 61, 108, 128, 158, 172, 228
  parameters 62
  receiver 228
  set 177
  status 107, 128, 158
  steer 46
  validity 128
CLOCKADJUST 46
CLOCKCALIBRATE 47
CLOCKMODEL 128
CLOCKSTEERING 130
CMR 69, 80, 159, 161, 189
  base station 189
  logs 132
CMRDATADESC 133
CMRDATAOBS 135
CMRDATAREF 137
cold start 140
COM 49
COM port 71, 73, 91, 139, 171, 224
COMCONFIG 135
COMCONTROL 51
communication commands 23
component types 234
configuration
  non-volatile memory 68
  port 23, 49, 139
  receiver 81, 105, 218, 220
  reset 32, 81
  save 90
  status masks 97
constellation 129, 172
constraints 177
control
  automatic 98
  channel 24
  commands 23
  filtering 106
  receiver 23, 105

conventions 10
coordinated universal time (UTC)
log 106, 108, 153–154
of position 141–142, 146
offset 158, 228
parameters 154
status 228
corrected mean motion 119
corrections
accuracy 65
bias 66
magnetic 75
magnitude of 74
RTCA 69
RTK 89
correlator spacing 179
corrupt data 77
CPU 71, 219
CRC, *see* cyclic redundancy check
cross track 148, 162–163
CSMOOTH 52
customer service 10
cut-off angle 61, 86, 172, 231
cyclic redundancy check (CRC) 13, 15–16, 21

D

data link 91
DATUM 52
datum 66
commands 24, 52, 56, 66
current 95
customized 102
ID 122, 157, 159, 165, 173, 214
transformation parameters 54
undulation 99
default
command tables 11
factory 23, 34, 53, 56, 81
destination 95, 163
de-weighting 70, 86, 172
DGPS command 56, 58–59, 69, 146
DGPSEPHEMDELAY 56
DGPSTIMEOUT 58
DGPSTXID 59
differential corrections
accept 69
age 58, 106–107, 122, 125–126, 165, 173–174, 176, 214–215, 217
fix position 65–66
none available 231
outages 122, 173, 214–215
positioning 56
send 78, 91
stations 66, 79, 88, 108, 231
transmit RTCA 56
dilution of precision (DOP) 106, 109, 141–142, 145, 172
direction
bearing 163
of motion 175
over ground 174
referenced to True North 74
distance
exceeded 124
straight line 163
track offset 95
dither 128
Doppler
assign 39–41
instantaneous 178
jump 46
range record 181
satellite visibility 226
tracking status 230
velocity computation 175
drift 46
DYNAMICS 60
dynamics 24, 66, 85

E

earth-centered earth-fixed (ECEF) 82–83, 126, 189
eccentricity 119, 150
ECEF, *see* earth-centered earth-fixed
ECUTOFF 61
elevation 147
cut-off 61, 172
error 231
highest 89
mask 86
RTK 26
satellite visibility 226
set 24
tracking status 230
ellipsoid

customized 102
height 82–83
navigation 95
parameters 53–54
surface 95
undulation 24, 99
environmental parameters 219
ephemeris
decoded 108
delay 56
health 231
log 109
raw data 108, 183
time status 19
errors
averaged position 78
clock 46, 62, 128
extrapolation 159, 161
flag 220
framing 171
in fixed coordinates 66
multipath 177
non-volatile memory 77
parity 171
range reject codes 231
response messages 258
status 97
text description 225
tracking 177
trigger 220
event
message 97, 220, 225
text description 225
trigger 220
type 225
expiry date 232
external oscillator 62
EXTERNALCLOCK 62
extrapolation error 159, 161

F

factory default
datum 53
ephemeris delay 56
modify 90
reset 23, 32, 81
settings 34, 49
field types 12
filter
control 24
error 231
pseudorange 230
RTK 24, 84
solution logs 106
filter update 175, 216
fine time 19
FIX 65
fix
command 65–66
data 141–142
first 147
position 143
save settings 90
types 66
fixed solution 24, 26, 84
flags
antenna 38
error 220
parity 177
status 173, 220
float solution 24, 26, 84
formats 13, 15, 21
framing errors 171
FREQUENCYOUT 67
FRESET 68

G

geodetic datum, *see* datum
geoid 24, 66, 99, 121, 141–142
GPGGA 142
GPGGARTK 141
GPGLL 143
GPGRS 144
GPGSA 145
GPGST 146
GPGSV 147
GPRMB 148
GPRMC 149
GPS overview 19
GPSEPHEM 150
GPSolution software 90
GPVTG 152
GPZDA 153
great circle line 95, 163

H

hardware
- parameters 219
- reset 23, 81
- version 105, 233

HDOP, *see* dilution of precision
header
- ascii 13–14, 18
- binary 12, 16, 18
- convention 11
- log 177

health
- almanac 140
- base station 108, 189
- satellite 226, 231
- status 150
- SV 119

height
- approximate 93
- baseline 83
- calculation 66
- ellipsoidal 83
- fix 24, 65–66
- limit 124
- mean sea level 82–83
- position 121–122, 157, 159, 165, 173, 214

hexadecimal 11–13, 16, 92, 97
hold 71–73, 155–156
horizon, cut-off angle 61

I

identifier
- ascii messages 13
- serial port 17, 69, 139, 171

inclination angle 119
initialization 82
interface 23, 25–26, 69
INTERFACEMODE 69
interrupt 171
ionospheric log 154
IONUTC 154

L

lane combinations 212
latched time 157
latency
- best position 122
- data link 159, 161
- position 106, 216
- velocity 107, 126, 175–176, 216–217

latitude/longitude
- approximate 93
- fix data 141–142
- GPS specific 149
- position 122, 143, 157, 159, 165, 173, 214
- RTK baseline 83
- set navigation waypoints 95
- sign 82

link
- data 91
- indicators 171

LNA, *see* low noise amplifier
local horizon 61
lock
- command 70
- loss of 66
- out 172, 231
- reinstate 100
- time 230

LOCKOUT 70
LOG 71
log
- list 155
- triggers 104
- types 104

LOGLIST 155
low noise amplifier (LNA) 23, 38, 219
LSB 13

M

magnetic variation 25, 74–75, 149, 163
MAGVAR 74
mark
- in event 106
- input pulse 157–158

MARKPOS 157
MARKTIME 158
mask
- angle 24, 86, 126, 161
- priority 97, 220–221
- RTK 25–26

matched update 212
MATCHEDPOS 159

matrix 128
mean sea level
  fix 66, 141–142
  position 122, 157, 159, 165, 173, 214
  RTK baseline 82–83
memory
  buffer space 71
  non-volatile 32, 68, 90, 119, 182
message
  almanac 119
  ascii 13
  base station 189
  formats 12, 15, 21
  ID 101, 155–156
  length 96
  navigation 95
  responses 18
  send 91
  trigger 72–73
MiLLennium commands 33
mode
  2-D 145
  3-D 145
  dynamics 85
  interface 69
  operating 145
  RTK 82, 122
  static 85
MODEL 76
model
  AUTH command 76
  authorization 23, 44–45
  card 76
  clock 128–129, 158
  expiry date 105, 232
  ionospheric 154
  log 106
  lost 77
  switch 76
  version 232–233
modem 170
month 228, 234
motion
  detector 85
  mean 119, 150
MSB 13
multipath 177

N

NAVIGATE 162
navigation
  commands 25
  data 145, 149
  information 148
  log 162
  magnetic compass 74
  path 95
  status 148, 162–163
  waypoint 95, 107
  word 185
NMEA 106–109, 142–143, 146
node 140
noise
  oscillators 62
  statistics 146
  thermal 177
  time of 129
non-printable characters 23, 92
non-volatile memory (NVM) 32
  erase 68
  reset 68
  restore 77
  save almanacs 119, 182
  save current configuration 90
note
  antenna motion 85
  authorization code 44
  channel assignment 39, 41
  clock adjustment 46
  differential corrections 56, 58, 65
  elevation cut-off angle 61
  ephemeris delay 56
  factory default 77
  GPSolution 90
  logging 71, 101
  navigation 162
  range residual 144
  reset 32, 68
  RTK 82, 84
  satellite 147, 172
  status 220, 225
  WGS84 121
NovAtel Inc. 2
NVMRESTORE 77

O

offset
- clock 61–62, 108, 128, 158, 172, 228
- Doppler 39
- track 95, 163

OMNIHPPOS 165
OMNIINFO 166
OMNISTAT 168
one pulse per second (1PPS) 26, 46, 67, 228
operating mode 145
optionality 11
origin 95
oscillator 46, 62, 128
output pulse 67
overload 71

P

parity 119, 171, 177, 183
PASSCOMx 170
pass-through logs 170
PDOP, *see* dilution of precision
perigee 119
period 71–73, 155–156
perpendicular distance 95, 163
phase lock loop (PLL) 178–179, 222
PLL, *see* phase lock loop
polled logs 104
ports
- ascii header 14
- COM 17, 23, 25–26, 49–50, 71, 73, 91, 224
- communication 15
- configuration 23, 49, 139
- number of interrupts 171
- output 72–73, 155–156
- pass-through 170
- receiver status 220
- save configurations 90
- serial 69, 171
- specify serial 69
- statistics 105
- status 171, 225
- unlog 101

PORTSTATS 171
POSAVE 78
position
- 3-D 172
- accuracy 65, 82
- at time of mark 157
- averaging 25, 78, 121
- base station 108, 189
- best 122, 126, 212
- commands 24
- current 162–163
- datum 53
- fix 24, 65–66
- logs 106–107
- pseudorange 108, 173
- solution 61, 172
- time matched 159, 161
- type 157
- xyz cartesian coordinate system 126, 161, 175, 216

power 38, 231
prerequisites 10
processing 14, 16, 108, 177
pseudorange
- corrections 66
- error estimate 177
- jump 46, 177
- measurement 146, 178, 181
- noise statistics 146
- position 107–108
- solutions 66, 122
- tracking status 230
- velocity 108, 174

PSRDIFFSOURCE 79
PSRPOS 173
PSRVEL 174
PSRXYZ 175
pulse, mark input 157–158

Q

quality 141–142, 146
quotation marks 13, 91, 96

R

RANGE 177
range
- bias 128–129
- compressed 180
- logs 177
- measurements 46
- reject codes 230–231

residual 144
satellite information 70
range bias 19–20
RANGECMP 180
raw almanac 140
RAWALM 182
RAWEPHEM 183
RAWGPSSUBFRAME 184
RAWGPSWORD 185
RAWOMNIFRAME 186
RAWOMNIPACKET 187
RAWWAASFRAME 188
reacquisition 39, 41, 179
receive
characters 171
interface 23, 26, 69
receiver status 71, 220, 223
reference station, *see* base station
references and standards 119, 182
REFSTATION 189
reinstate satellites 100
reject code 231
remote station, *see* rover station
RESET 81
reset
average positions after 78
complete 84
hardware 23, 68, 81
residual 144, 212, 230
resolution 82, 84
response 18, 69, 258
restore 77
right ascension 119
RMS, *see* root mean square
root mean square (RMS) 146
rover station
commands 26
logs 108
RTCA 56, 58, 69, 80, 159, 161, 189
base station 189
logs 190
RTCADATA1 191
RTCADATAEPHEM 192
RTCADATAOBS 193
RTCADATAREF 195
RTCM 56, 69, 80, 159, 161, 189
base station 189
logs 196
RTCMDATA1 197
RTCMDATA15 200
RTCMDATA16 201
RTCMDATA1819 202
RTCMDATA2021 206
RTCMDATA3 198
RTCMDATA59 210
RTCMDATA9 199
RTK
baseline 24, 26, 82, 108
commands 26
conventions 10
corrections 89
data 108, 212
filter 84
information 213
low latency position 106, 214
mode 122, 216
observations 86
position 106, 122, 159, 161, 214
processing 87
satellite counts 108
solution 87, 212
velocity 215
RTKBASELINE 82
RTKCOMMAND 84
RTKDATA 209
RTKDYNAMICS 85
RTKELEVMASK 86
RTKPOS 214
RTKSOLUTION 87
RTKSOURCE 88
RTKSVENTRIES 89
RTKVEL 215
RTKXYZ 216
RXCONFIG 218
RXHWLEVELS 219
RXSTATUS 220
RXSTATUSEVENT 225

S

satellites
acquisition 39
active 145
almanac 119, 140
available 66
clock dither 62
commands 24

common 212
count 108
elevation cut-off 61
errors 231
geometry 172
GLONASS slot number 178, 226, 230
in view 147
lock 70, 172
number of 172, 180
PRN 178
range 144
raw 182–185
reinstate 100
RTK 25, 86, 89, 212
tracking 109, 177, 230
unassign 98
unlock 100
visibility 109, 226
SATVIS 226
SATXYZ 227
SAVECONFIG 90
SBAS 10–11
scaling
almanac 119
factor 181
scope 10
searcher status 213
self-test 71
SEND 91
send 91–92
SENDHEX 92
serial port 69, 139, 171
SETAPPROXPOS 93
SETAPPROXTIME 94
SETNAV 93
software version 105
solution status 124, 157
speed
current 163
data 106, 149
over ground 125, 149, 152, 174, 215
SPS, *see* standard positioning service
standard positioning service (SPS) 119, 182
standards and references 119, 182
static mode 85, 161
station ID 59, 79, 88, 189
status
arrival 148
auxiliary 224
base station health 189
channel tracking 177–178, 181, 230
clock model 158
COM ports 171
commands 23
data 143
event 225
flag 173, 220
indicators 122, 125–126, 175, 216
mask 97
navigation 162
receiver 14, 71, 105, 220, 223
self-test 105
solution 124, 157
time 14, 19
velocity 126, 215
word 225
STATUSCONFIG 97
steer
clock 46
time 19–20
subframe 109, 119, 182–183
synchronous logs 104

T

TIME 227
time
almanac reference 182
anomaly 119
clock adjustments 46
coarse/fine 19
CPU 71
data 228
delay 56
dilution of precision 172
ephemeris delay 56
fine 19
GPS 128, 228
improve acquisition 65
interval 72–73, 175
latched 157
limit 78
logs 107, 228
matched position 108, 161
of ephemeris 183
of first fix 147

of mark in event 158
of position fix 143
precision 19
stamp 161
status 14, 19
steering 19, 46
tag 175, 189, 216
timeout 58
UTC 141–142, 153–154, 228
validity 19
TIMESYNC 229
track
made good 149, 152
over ground 125, 174, 215
tracking
assign 39
automatic 98
channel 177, 230
continuous 168, 178, 230
cut-off angle 61
fix position 66
loops 177
satellite 24, 109
status 179, 230
SV 172
undesirable 70
TRACKSTAT 230
transformation parameters 53
transmit 49, 171
interface 23, 26, 69
trigger
error 220
event messages 97
log 71, 104, 155–156, 159, 161
option 73
true north 74–75, 125, 146, 163, 174, 215
types 12

U

UNASSIGN 98
UNASSIGNALL 98
UNDULATION 99
undulation
best position 122, 165
command 82, 99
height 82
illustration 99
position 121, 157, 159, 173, 214
UNLOCKOUT 100
UNLOCKOUTALL 100
UNLOG 101
UNLOGALL 101
USERDATUM 102

V

validity
base station 189, 216
clock model 128
receiver model 232–233
time tag 175, 216
VALIDMODELS 232
VCTCXO, *see* oscillators
VDOP, *see* dilution of precision
velocity
average 175, 216
best 125–126
closing 148
fix 24, 65–66
latency 175, 216–217
limit 124
logs 107
pseudorange 108
RTK 215
status 175
xyz cartesian coordinate system 126, 175, 216
VERSION 233
version 2, 44, 105, 233–234
virtual address 14
visibility, satellite 109, 226
voltage 219
Volume 1
operation 214–216
prerequisites 10
pseudorange algorithms 121
RTK 132–133, 135, 137, 190–193, 195–202, 206, 209–210
standards and references 119, 182
status tests 220
technical specifications 157–158, 225
unit conversion 178

W

WAAS0 235
WAAS1 236
WAAS10 247

WAAS12 248
WAAS17 249
WAAS18 250
WAAS2 237
WAAS24 251
WAAS25 253
WAAS26 255
WAAS27 256
WAAS3 239
WAAS4 240
WAAS5 241
WAAS6 242
WAAS7 244
WAAS9 246
WAASCORR 257
WAASCORRECTION 103
warning 44
warranty 10
waypoint
    destination 148, 163
    navigation 25, 95, 107, 162
    track offset 95
week
    future 154
    GPS 150, 163
    reference 182–183
weighting, pseudorange filter 230
WGS84 53, 55, 66, 95, 99, 102, 126
word 77, 97, 140, 183, 220, 225

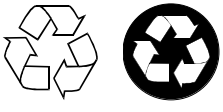
X

xyz cartesian coordinate system 106–107, 126, 175, 189

Y

year 228, 234

Recyclable



Printed in Canada on recycled paper